IIAA99
Containment Model and Database Technical Documentation

Prepared by

Alan D. Larrabee

FedSource
Contract Number 6075
Task Number NW02-286

Alan D. Larrabee, Inc.
P.O. Box 16614
Albuquerque, NM 87191-6614

(505) 294-1258
ConsultADL@aol.com

Introduction

This is the final report for the FPA Project IIAA Technical Documentation task. The work was performed for the Bureau of Land Management by Alan D. Larrabee, Incorporated, under FedSource Contract Number 6075 and Task Number NW02-286. The objective of this task order was to create technical documentation of those portions of the IIAA Computer program and its associated database that may potentially be incorporated into the software being developed for FPA.

The deliverables under Task Order NW02-286 consist of this report and a series of files contained on the CD-ROM entitled "FPA Project, IIAA Technical Documentation, 10/31/02". This CD-ROM includes the following information:

Item	Type	Content
SourceCode	Folder	IIAA source files that were documented
iiaa99.new	File	IIAA template database that was documented
FinalReportNW02-286.doc	File	This report, in Microsoft Word 2002 format
IIAA99.vsd	File	Program flow diagram, in Microsoft Visio 2002 format
IIAA99dotMDB.vsd	File	IIAA database diagram, in Microsoft Visio 2002 format
Procedures.mdb	File	Annotated cross reference of procedures contained in IIAA99, in Microsoft Access 2000 format

Table 1

The IIAA Program

IIAA, or Interagency Initial Attack Analysis, is a component of the NFMAS Suite of programs. The program has evolved over many years of service, and may currently be considered to be legacy code. In its present form, the program runs on IBM PC platforms under all forms of the Microsoft Windows operating system from Windows 95 forward. The Windows versions of the program were implemented and continue to be maintained by Bighorn Information Systems, Costa Mesa, California.

IIAA is programmed in the Microsoft Visual Basic language. The original Windows version of the code was developed using Visual Basic 3 and Access 95. In 1999, the program and its associated database were converted from a 16-bit format to the 32-bit format using Visual Basic 6 and Access 97, then renamed IIAA99. This report documents Version 1.2.9 of the IIAA99 source program.

A developer who attempts to compile IIAA99 must have several additional components. IIAA99 produces two different graphs, a graph of AAC values and a stacked bar chart representing the classic Sparhawk curve. IIAA99 requires the Microsoft Chart 1.0 control to generate these graphs. Visual Basic 6 uses MS Chart 2.0. A developer who does not possess a licensed copy of MS Chart 1.0 must convert IIAA to MS Chart 2.0 to successfully compile the program.

All of the tables displayed on IIAA99 forms were programmed using the TrueDBGrid Pro 5.0 control. Apex Software, the company which originally authored this third-party software has been acquired by ComponentOne, Inc. ComponentOne is currently marketing Version 7.0 of this control. A developer who does not possess a licensed copy of TrueDBGrid Pro 5.0 must convert IIAA to Version 7.0 of this control to successfully compile the program. Since this control appears on more than 30 forms within IIAA99, this would not be a trivial task. Alternatively, it may be possible for users of TrueDBGrid Pro 7.0 to purchase prior versions of the control from ComponentOne at additional cost.

The file containing the IIAA source code includes four different Visual Basic projects: copysrc, ia5100-2, iiaa32, and vb3to6. These projects serve the following purpose:

Project	Function
copysrc	Creates a MS DOS batch (.BAT) file which, when executed, will copy all forms and code modules contained in the FORBS project folder (not supplied) to a "COPYSRC" subfolder.
ia5100-2	Produces the USFS 5100-2 and IIAA Rollup.
iiaa32	The actual IIAA99 program, which simulates the effectiveness of various Initial Attack organizations or options for a particular Planning Unit.
vb3to6	A utility program which modifies the syntax of certain SQL statements associated with the specified VB forms. This program apparently was used during the conversion of IIAA from Visual Basic Version 3 to Version 6.

Table 2

Only the iiaa32 Visual Basic project containing IIAA99 was documented under this task order. Further, due to the limited time frame of the task order, only those IIAA99 components associated with the actual containment model were documented.

Program Metrics

The IIAA99 source is comprised of 61 forms and 9 code modules. Contained within the forms and code modules are 944 individual procedures of which 119 are unreferenced within the project. The source code consists of the following items:

Line Type	Count
Actual Code	21,823
Comment	830
Blank	4,460
Unreferenced ("Dead" Code)	4,610
Total	31,723

Table 3

Variable Scope	Count
Local	2,605
Module	204
Global	108
Total	3,017

Table 4

Further information regarding empty or unreferenced procedures and the use of classes may be found in Appendix A of this report.

Although there are more than 800 comment lines in the code, this figure is misleading. Most of the "comment" lines are either debug statements that have been commented out, or code that was functional in the 16-bit version of IIAA99 and which has been left in the current version as comments after the conversion to the 32-bit version. There are very few actual comment lines (in the traditional sense) contained in the source code. Thus, while judicious use of blank lines has enhanced the readability of the code, IIAA99 is essentially uncommented.

IIAA99 contains 119 unreferenced procedures consisting of more than 4,600 lines of code. Although it appears that approximately 15% of the total program is "dead code", a nontrivial percentage of the unreferenced procedures are contained in various utility modules. These procedures are probably used elsewhere in the NFMAS Suite of programs, but not used by IIAA99. This does not negate the fact, however, that a significant portion of IIAA99 contains code that is completely unreferenced. An example of this situation would be the A, B, and C summary and detail reports which apparently were dropped at some point. The programming that generates these reports still remains in the source code, but is never called.

In addition to the obviously unreferenced code, another problem is code which has been functionally disabled. These code lines have been left in the source, but have been "bridged around" by various means so that they will never be executed. The Size Up capability is an example of this approach. When the Size Up function was dropped in past versions of IIAA99, the code was not removed from the system. Rather, IIAA99 is initialized with a set of conditions which prevents execution of this portion of the code. Programmers examining the IIAA99 source may occasionally experience some difficulty in ascertaining whether a particular section of this type of code ever gets executed.

IIAA99 Program Documentation

The technical documentation of the IIAA99 program presented in this report consists of three parts: Key program module identification and description, diagrams of the program logic flow between these modules, and a cross-reference of all procedures in the IIAA99 Visual Basic project. The logic flow diagram, implemented with Visio 2002 Professional, is found on the CD-ROM companion to this report. The flow diagrams for modules comprising the IIAA Suppression Model have been extracted from the Visio file and are presented in Appendix B of this report.

Because it documents the procedure calls and references for all 944 IIAA99 procedures, the procedure cross-reference was considered too large to be included as a printed Appendix to this report. This cross-reference may be found on the companion CD-ROM as a Microsoft Access database. The actual cross-reference is contained in the table "Procedures", which may be sorted by procedure name within module, or by procedure name alone. For a given module and procedure, calls and references to the procedure will be found respectively in the Memo fields "Calls" and "References".

The procedures that comprise the IIAA99 simulation and containment model are identified and described in the following table:

Object	Object Type	Procedure	Description
frmRun	Form	BuildOSTs	Retrieves CDPCode from ProgOpts. If Option uses a Consistent Dispatch Philosophy, then call IIAA:CalcCDP to build OST, otherwise call IIAA:CalcFIL to load default FILs from Line Items table.
frmRun	Form	intCheckRun	Verifies that user has selected FBD, FMZ's, Options, and that FBD data exists for all selected FMZ's
IIAA	Module	AutoMRT	Build and replace records in the Master Resource Table for those Line Items flagged as UseAutoMRT = True.
IIAA	Module	CalcFIL	Build OST using default FILs from Line Items table for resources included in an Option.
IIAA	Module	CalcNVC	Updates the NVC Table.
IIAARun	Module	CheckEnd	Examines fire perimeter vs line built. Examine fire size & elapsed time vs. size & time escape constraints. Set flag if fire caught or escaped.
IIAARun	Module	FetchArrivals	Finds: [Arrival time of 1st Watertender] OR [1st Ground Resource (Engine or Watertender)] AND [1st ground resource with water.]
IIAARun	Module	HandleMRT	Determines production rate of new resource, then updates current total production rate.
IIAARun	Module	HandleAT	Updates total line built with contribution from previously dispatched aerial resources.
IIAARun	Module	HandleRefill	Adjusts production rate to account for the effect of refills i.e., ground resources which have departed to refill or which have arrived after refilling.
IIAARun	Module	HandleChildren	Builds OST entries for Line Items created by IIAA (as opposed to those entered by user) and related to primary Line Items included in an Option (e.g., 2nd and subsequent helitack squads and water drops for helicopters). NOTE: This procedure has been coded to allow the manipulation of generalized "Target Tables" containing entries which can be linked to records in the Line Items table where the Parent and/or Child ID fields contain a LineID. However, all calls to HandleChildren from within IIAA specify the OST as the Target Table. Therefore, the net effect is that HandleChildren maintains only the OST.
IIAARun	Module	InitRun	Initializes variables for specified ROS. Writes chronological Event Log Header.
IIAARun	Module	intCatchOrEscape	Determines whether a fire has been caught or escaped. Calls HandleAT, HandleRefill, and CheckEnd. Calculates total line built.
IIAARun	Module	intRun2	Runs the actual model for specified FMZ/Option.
IIAARun	Module	intFetchEFT	For a specified FBD, gets Escaped Fire Size thresholds for each RL/FIL at ROS50 and ROS90.
IIAARun	Module	intFetchFBD	For a specified FBD, gets Annual Fire Frequency and Rate of Spread at the 50th and 90th percentiles for each FIL.

Object	Object Type	Procedure	Description
IIAARun	Module	intFetchFMZ	Gets Fuel Model, RL Count, Escape Acres, and Escape Minutes for a particular FMZ.
IIAARun	Module	intFetchRunNum	Performs maintenance functions for tables: Runs, RunTable5A and RunTable14. Increments run number and returns new value for a specific FMZ & Option. (Note: Run Number resets to 1 when it exceeds 32000).
IIAARun	Module	intFMZRL	For each RL in the specified FMZ, gets the percentage of total fires, Water Minutes, Airtanker Reload Minutes, and Walk-in Delay Minutes.
IIAARun	Module	RecordEvent	Writes Event Log entries and creates records in RunTable14 for fires which have escaped due to size or time constraints or which have been suppressed..
NFMASRun	Module	sngAcres	Single-precision function that returns fire size in acres as a function of elapsed time, ROS, and Discovery Size.
NFMASRun	Module	sngAirDropLine	Single-precision function that returns line built by this drop, as a function of FIL, ROS, Fuel Model and resource capacity.
NFMASRun	Module	sngPerim	Single-precision function that returns fire perimeter in chains as a function of elapsed time, ROS, Fuel Model, FIL, and Discovery Perimeter.
REPORT1	Module	AddFreq	Loads information into User Deftined Type “Dispatch” for use in generation of dispatch reports for a specified FMZ/Option/FBD.
REPORT1	Module	DispatchReport	Prints Dispatch Report for a specified FMZ/Option/FBD.
REPORT1	Module	sngRunFreq	Single-precision function called by AddFreq that maintains a running sum of mission frequency for a given resource (Line Item) for use in generating the Suppression Table 5 report.
REPORT1	Module	SuppTable5	Prints Suppression Table 5 for a specified FMZ/Option/FBD.

Table 5

IIAA99 Database Documentation

IIAA99 stores all of its information in a Microsoft Access database. Because this database is referenced from within IIAA by DAO Object Library 3.51, it must always be in Access 97 format. The database consists of 58 tables and 6 queries or stored procedures. It contains no forms or reports, since these functions are performed from within IIAA99. The IIAA99 database contains a version number, which is used in conjunction with the build number compiled into the IIAA99 executable to specify the current version of IIAA. The database version number is found in the “Param” table associated with the Key: ParamCode = dbVer. The database version documented by this report is IIAA99 Database Version 15. Due to ongoing modifications made by Bighorn Software, the current version of the IIAA99 database, as of the date of this report, is thought to be Version 17.

The IIAA99 database is fielded in the IIAA99 Setup file, and must always be installed in the same folder as the IIAA99 executable, and is named “iiaa99.new”. Each individual Planning Unit requires its own customized copy of the database. Thus, the “dot new” file is actually a template which is used in

conjunction with data specific to an individual Planning Unit to create the production database. In its "dot new" form, most of the tables in the database are empty except for tables containing information preloaded by the software developer. The IIAACodes, InflateFactors, and Param tables are preloaded with generic information applicable to all types of planning units. The Color (Budget Code) and CostCategory tables are preloaded with USFS codes. These codes are replaced with the codes appropriate for BIA and BLM planning units during the database customization process, which occurs when a user selects the "New Database" option from the File menu. The customized database thus produced follows the file naming convention: IIAA99XXXX … XX.MDB where XXXX … XX is a user defined text string that allows identification of the planning unit or content of the database. Unlike the "dot new" template, the production IIAA99 database may reside anywhere on the hard drive.

Although the database was constructed using a relational model, no formal relationships between tables are defined within the database itself. The IIAA99 program has the capability to display and print the database schema from the "Database Structure" option of the Utilities menu. The ""Database Browser" option of the Utilities menu allows a user to display and print the content of any table in the database. However, no generalized query generator is provided and all reports must be produced from the choices available as various menu options from within IIAA99. The IIAA99 database is intended to be a single-user database with full access to all information contained therein. A "power user" with a copy of Microsoft Access could open the database and either run queries against it or produce reports. If, during such a process, information within the database was altered in a manner inconsistent with IIAA99, obvious instability could result.

The technical documentation of the IIAA99 database presented in this report consists of two parts: A description of each of the tables and queries contained in the database, and a series of diagrams defining the relationship between the various tables. The relationship diagram, implemented with Visio 2002 Professional, is found on the CD-ROM companion to this report. These diagrams have also been extracted from the Visio file and are included in Appendix C of this report. A description of the objects in the database is presented in the following table:

Object	Type	Visio Tab	Description
AAC	Table	Unit	Holds default breakpoint and Average Acre Cost data for each FMZ.
AACExcept	Table	Unit	Exceptions to the Average Acre Cost data stored in Table AAC, organized by RL within FMZ. This data will override the default AAC.
BaseLocations	Table	Unit	Initial Dispatch Location IDs and descriptions.
BaseRLMiles	Table	Unit	Mileage from each base to each FMZ/RL.
CDP	Table	CDP	Consistent Dispatch Philosophy codes and descriptions.
CDPExcept	Table	CDP	For Options that use a Consistent Dispatch Philosophy, this table stores alternate CDP's by Option/FMZ/RL which are used to override the default values
CDPLine	Table	CDP	Consistent Dispatch Philosophy Line Item Group IDs and descriptions. CDPLine:CDPLine is inserted into the CDPLine field in LineItems and LineIDPart records as a foreign key.
Color	Table	Budget	Holds Color (Budget Account) codes and descriptions, as well as flags that control how budget amounts are calculated.
Coop	Table	LineItems	Cooperator IDs and Descriptions, plus production rate and delay information used by the containment model.
CostCategory	Table	Budget	Cost Category codes and descriptions.

Object	Type	Visio Tab	Description
Dispatch	Table	CDP	For each Consistent Dispatch Philosophy (CDPCode) and Resource Group (CDPLine), stores additional resources to dispatch both “from our unit” and “from all Units” by FIL.
EFT	Table	Unit	The Escaped Fire Table stores fire escape size in acres by FMZ/RL/FIL/ROS for a particular FBD.
FBD	Table	Cal	The Fire Behavior Data Table stores fire frequency and rate of spread in chains per hour at both the 50th and 90th percentile by FMZ/FBD/FIL.
FBDTitle	Table	Cal	Fire Behavior Data Codes (typically “HIS” and “CAL”) and descriptions.
FCO	Table	OptBust	Fire Control Objective IDs and definition data.
FCOAssign	Table	OptBust	Assigns Fire Control Objective IDs to specific FMZ/RL/FILs.
FCODetail	Table	OptBust	For each selected Option, this interim Option Buster output table stores the percentage of fires contained by FCO/FMZ/RL/FIL. Contents of this table are used to create the detail section of the Option Buster report.
FCOOpt	Table	OptBust	For each selected Option, this interim Option Buster output table stores the FCO Pass/Conditional Pass/Fail Status. Contents of this table are used to determine budget levels, then find the “best option” with each budget level.
FCOResults	Table	OptBust	For each assigned FCO for selected Options, this interim Option Buster output table stores the number of annual fires both above and below the FCO containment goal (expressed as a percentage of fires) and the Pass/Fail status.
FMZ	Table	Unit	FMZ Codes, descriptions, and basic information for each Fire Management Zone within the Planning Unit.
FMZRL	Table	Unit	For each RL within each FMZ, this table stores the percentage of fires for the RL, various reload and delay times used in the dispatch of resources, and the Latitude/Longitude coordinates of the RL.
HFT	Table	Cal	The Historic Fire Table, created by manual edit or, typically, by importing data downloaded from the Shared Application Computer System (or the SACS CD-ROM) and formatted for IIAA by PCHA.
IIAACodes	Table	LineItems	Codes and descriptions for each possible type of Line Item, organized by the Agency to which a specified Planning Unit belongs. Code Set 1 applies to the Bureau of Land Management (BLM), Code Set 2 applies to the USDA Forest Service, and Code Set 3 applies to the Bureau of Indian Affairs (BIA).
InflateFactors	Table	Budget	Multipliers used to inflate Budget, AAC, UMC, and NVC amounts from their respective Budget Base Years to a specified Target Year.
LineCat	Table	Budget	Uninflated budget amounts associated with specific Line Items/Color (Budget) Codes/Cost Categories. May be either default amounts or “Special Cost” amounts entered from the Included Items List for a specified Option.

Object	Type	Visio Tab	Description
LineIDPart	Table	LineItems	Information about Producer Types, used as default values when entering Line Items or building the Master Resource Table (MRT).
LineItems	Table	LineItems	Information about specific Budget Line Items used by the Master Resource Table (MRT), Option Selector Table (OST), Included Items Table (OptLine). And the IIAA model.
LinePers	Table	Budget	Personnel information for Planning Unit employees attached to a specific Line Item. This table is indexed, but has no primary key(s). Key structure should be LineID/Color/Rownum. Under certain conditions, data records may become “orphaned”.
ListReport	Table	Misc&Model	If the user has created and saved Custom Reports, this table stores the Report Names and the associated SQL query.
MRT	Table	LineItems	The Master Resource Table is organized by LineID/FMZ/RL and stores data used a default values for Line Items or for use by the IIAA model.
NVCImmature	Table	NVC	Organized by Immature ID, Species, and Pole or Seedling/Sapling Code, this table stores child data for NVCImmatureID.
NVCImmatureID	Table	NVC	ID Code (Stand Identifier) and description for the Immature Timber worksheet.
NVCMature	Table	NVC	Organized by MatureID and Species, this table holds the child data describing mortality and non-salvage value by FIL for NVCMatureID.
NVCMatureID	Table	NVC	ID Code (Stand Identifier) and description for the Mature Timber worksheet.
NVCParam	Table	NVC	This table contains the NVC Table IDs, descriptions, and stand sizes in acres together with foreign keys to the Mature and Immature timber tables.
NVCResourceLookup	Table	NVC	This table provides child data to NVCWorksheet, and is preloaded with the “canned” Resource Categories processed by IIAA.
NVCSetup	Table	NVC	Contains data which assigns a NVC Table to each FMZ/RL defined for the Planning Unit.
NVCStump	Table	NVC	Supplies timber species and associated stumpage values as child data to the NVCMature and NVCImmature tables.
NVCTable	Table	NVC	Contains the final weighted NVC value (per acre) calculated by IIAA just prior to running the model or upon exit from the “Final NVC Values” form.
NVCWorksheet	Table	NVC	This table contains child data for NVCParam and is used in the Immature Timber worksheet.
OptBustSummary	Table	OptBust	This table holds data calculated for use in generating the Summary section of the Option Buster report.
OptLine	Table	LineItems	Commonly referred to as the “Included Items List”, this table specifies the Line Items associated with each Program Option.
OST	Table	LineItems	Organized by Line Item/Option/FMZ/RL, the Option Selector Table specifies the minimum FIL at which each “producer” resource may be dispatched.

Object	Type	Visio Tab	Description
Param	Table	Misc&Model	Contains system parameters such as Unit ID, Unit Name, Budget Base Years and is used by many different procedures within IIAA.
Personnel	Table	Budget	Pay rates, Appointment Types, and Employee Benefit Cost information in support of the LinePers Table. The Personnel table is not keyed. The Primary key should be "PersID". Storing duplicate pay rates (records with the same PersID but differing cost, status or EBC can cause inconsistent personnel cost calculations for Line Items or in Rollup)
PersStatus	Table	Budget	Status codes and descriptions for Appointment Types. This table supplies child data for the Personnel Table.
PRates	Table	LineItems	This table holds default production rates in chains per hour by NFDRS Fuel Model for the Producer Types stored in LineIDPart.
ProgOpts	Table	LineItems	For all Program Options defined in IIAA, this table stores their Option ID, description, and Consistent Dispatch Philosophy Code, if applicable.
ReportGroups	Table	DeadTables	A dead table formerly used by dead code module REPORT2 to create the "A" Report (Line Items by Option), ""B" Report (Line Items by Color Code), and the "C" Report (Line Items by Cost Category). Both Summary and Detail reports were produced. These reports were disabled during the conversion of IIAA to the 32-bit version.
ReportLines	Table	DeadTables	See ReportGroups
ReportTitles	Table	DeadTables	See ReportGroups
RunData	Table	Cal	This table is the output of the calibration process. Table RunData is not keyed. RunData:OptID maps to FBD:FBDCode and RunData:FMZ maps to FBD:FMZCode. FBD:Freq1 through FBD:Freq6 and FBD:ROS1xx through FBD:ROS6xx correspond to RunData:FIL 1 through 6 respectively.
RunGroups	Table	LineItems	This table contains the Run Group ID and description used to model 2 or more Program Options as a "group" without forcing the user to specify the individual Option IDs. NOTE: RunGroups:Code maps to ProgOpts:RunID.
Runs	Table	Misc&Model	This table hold the Run Numbers (the ordinal generated by IIAA whose value is between 1 and 32,000) and run dates associated with each FMZ/Option/FBD.
RunTable14	Table	Misc&Model	Containing the output of the IIAA model, this table stores the contained/escaped status, Fire Frequency, Fire Size, and Line Built by Run Number/RL/FIL for both the 50th and 90th percentile Rate of Spread (ROS).
RunTable5A	Table	DeadTables	This table has been effectively disabled, although it is referenced should the Run Number "roll over" from 32,000 to 1. For a particular Run, it appears to have stored the annual Fire Frequency by Size Class and/or the Dispatch Frequency for a specified Producer Type, and was probably used to produce a report in the "Suppression Table 5" family.

Object	Type	Visio Tab	Description
SizeUp	Table	Unit	Holds the Size Up delay for each FIL. NOTE: The Size Up capability has been effectively disabled by IIAA setting the delay values to 0 for FIL 1 through FIL 6.
SubUnits	Table	LineItems	This table contains child data for LineItems. It holds the Sub Unit Code and description for sub units of the main Planning Unit e.g., a particular Ranger District with a National Forest. Although the Sub Unit Code is carried as a foreign key in each Line Items record, IIAA appears not to use this information in any of its reports.
FirstGround	Query	Queries	For a specified Option, finds the arrival time of the first ground unit to respond to a fire at a given FMZ.RL/FIL.
FirstWT	Query	Queries	For a specified Option, finds the arrival time of the first water tender to respond to a fire at a given FMZ.RL/FIL
LineBudget	Query	Queries	For a specified Option and Line Item ID, calculates the budget associated with Line Items which contribute to the total C Plus NVC.
NonBudget	Query	Queries	For a specified Option and Line Item ID, calculates the budget associated with Line Items which do not contribute to the total C Plus NVC.
QDRunData	Query	Queries	Retrieves information for use during the calibration process.
RunStuff	Query	Queries	For a specified Option/FMZ/RL, retrieves MRT and Line Item resource data for use by the IIAA Model.

Table 6

APPENDIX A

IIAA99 Program Information

APPENDIX A-1

Empty Procedures

The following 21 procedures contain no executable code:

Procedure	Module
cmdSave_Click(Event)	frmMRTP
Label34_ClickEvent)	LI32
LaunchImage(Method)	IIAA
mnuColor_Click(Event)	MDIForm1
mnuDCat_Click(Event)	MDIForm1
mnuInflate_Click(Event)	MDIForm1
mnuPersT_Click(Event)	MDIForm1
mnuProdType_Click(Event)	MDIForm1
mnuReportGroupsItem_Click(Event)	MDIForm1
mnuReportLinesItem_Click(Event)	MDIForm1
mnuReportTitlesItem_Click(Event)	MDIForm1
mnuSpread_Click(Event)	MDIForm1
mnuTestAReport_Click(Event)	MDIForm1
mnuUtilDrop_Click(Event)	MDIForm1
mnuWONVC_Click(Event)	MDIForm1
ShowMsg(Method)	frmSetup
ShowMsg(Method)	frmDates
ShowMsg(Method)	frmMRTP
sRTFBaseBox(Method)	B3Commonia
TDBGrid1_Click(Event)	Stumpage
txtLineID_GotFocus(Event	LI32

APPENDIX A-2

Unreferenced Procedures

Procedure	Module	Lines
ACOS	B3Commonia	3
ADetailReport	REPORT2	143
AReport	REPORT2	123
BDetailReport	REPORT2	165
BReport	REPORT2	148
CDetailReport	REPORT2	166
CenterMe	B3Commonia	4
CReport	REPORT2	147
DD2DMS	B3Commonia	18
DMS2DD	B3Commonia	6
fContinue	dlgReports	45
FileCenter	B3Commonia	4
g1_AfterColEdit	FMZRL	12
g1_AfterColEdit	frmIILS	18
G1_AfterColUpdate	Stumpage	12
G1_AfterColUpdate	Immature	11
g1_BeforeColEdit	BaseMiles	8
g1_BeforeColEdit	FMZRL	6
g1_BeforeColEdit	FBD	6
g1_BeforeColEdit	OST	8
g1_BeforeColEdit	Stumpage	8
g1_BeforeColEdit	defPRate	6
g1_BeforeColEdit	mrt	15
g1_BeforeColEdit	IIL	8
g1_BeforeColEdit	HFT	7
g1_BeforeColEdit	Disp32	7
g1_BeforeColEdit	Immature	10
g1_BeforeColEdit	frmNVCTot	10
g1_BeforeColEdit	Mature	10
g1_BeforeColEdit	DefFCO	9
g1_BeforeColEdit	DefMO	9
g1_RowColChange	frmNVCSetup	18
g1_RowColChange	IIL	36
g1_RowColChange	frmCal2	10
g1_SelChange	frmNVCSetup	6
g2_AfterColUpdate	LI32	31
g2_AfterColUpdate	Immature	7
G2_AfterDelete	LI32	14
g2_BeforeColEdit	frmNVCSetup	15
g2_BeforeColEdit	LI32	8
g2_BeforeColEdit	Immature	11
G2_BeforeDelete	LI32	10
G2_SelChange	LI32	6
g3_AfterColUpdate	LI32	23
g3_AfterDelete	LI32	29

Procedure	Module	Lines
G3_BeforeColEdit	frmNVCSetup	8
g3_BeforeDelete	LI32	10
g3_SelChange	LI32	6
GetParamDate	B3Commonia	12
Grid1_HeadClick	DBBrowse32	21
HighlightSelect	B3Commonia	12
iCoopNulls	IIAA	11
iCountChar	B3Commonia	13
iCreateNewMDB	B3Commonia	52
iDimGridArrays	GridMod	69
iFillResources	OST	23
iJulian365	B3Commonia	38
iLoadAff	Immature	19
iLoadMe	BaseMiles	60
iLoadMe	FMZRL	82
iLoadMe	FBD	58
iLoadMe	frmNVCSetup	60
iLoadMe	IIL	182
iLoadMe	frmIILS	87
iLoadMe	HFT	36
iLoadMe	Disp32	42
iLoadMe	Immature	85
iLoadMe	Mature	52
iLoadMe	AAC	50
InputTables	REPORT1	17
Instructions	frmLID	13
Instructions	LI32	13
intCreRunDataTable	NFMAS2b	47
intOKFM	B3Commonia	9
iOldDBUpdates	IIAA	365
iPrintCustomSQLDB	B3Commonia	124
iRemoveOrphans	B3Commonia	14
iRemoveOrphans2	B3Commonia	14
iResetPrimary	B3Commonia	16
iRO	MDIForm1	10
iSaveAff	Immature	16
iSaveGrid	mrt	47
iSaveMe	BaseMiles	25
iSaveMe	FBD	23
iSaveMe	OST	31
iSaveMe	mrt	30
iSaveMe	IIL	42
iSaveMe	HFT	34
iSaveMe	Disp32	36
iSaveMe	AACrl	29
iSaveMe	AAC	26
iShowStructure	B3Commonia	87
iSuppTable1	REPORT1	130
iSuppTable1Header	REPORT1	11
iSuppTable2	REPORT1	146
iSuppTable2a	REPORT1	135
iSuppTable2Header	REPORT1	9
iTestOpen	B3Commonia	22
Jul2String	B3Commonia	35
LF2CRLF	B3Commonia	67

Procedure	Module	Lines
lID	B3Commonia	15
lngAAC	IIAARun	28
MakeDB	B3Commonia	6
OpenDBFile	B3Commonia	49
PrintLong	B3Commonia	22
RightForm	B3Commonia	17
sCode	B3Commonia	15
SetListSelect	B3Commonia	9
ShowImage	IIAA	16
sJulian1900	B3Commonia	40
sngSqFtToAcres	B3Commonia	4
sRTFBase	B3Commonia	8
sRTFBaseBox	B3Commonia	3
strLstWhere	B3Commonia	30
strParseQuery	B3Commonia	52
T5ACount	IIAARun	27
Timer1_Timer	DARTSchk	33
Timer16_Timer	MDIForm1	136
WriteParamDate	B3Commonia	13

Total Lines Unused In This Project: 4610

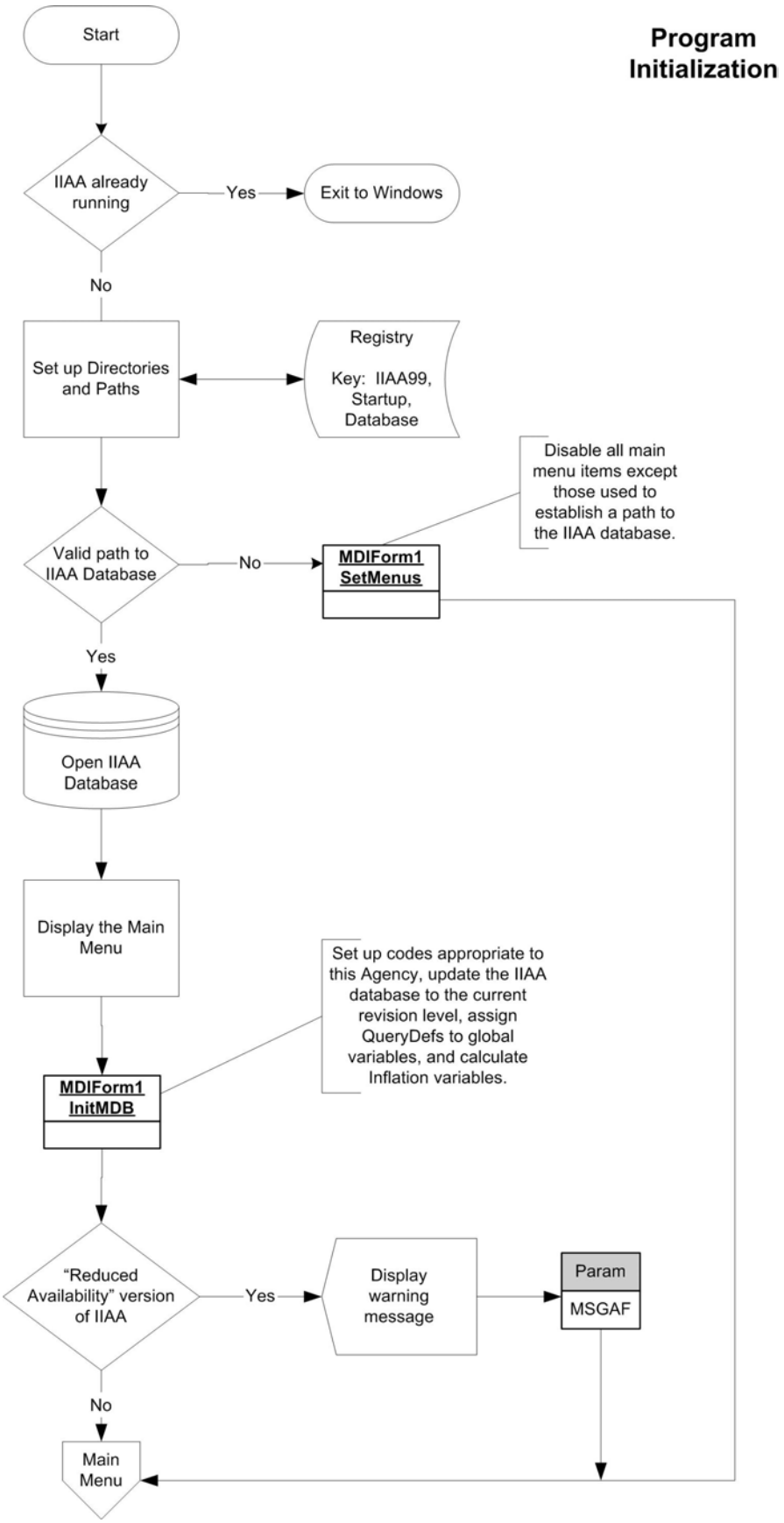
APPENDIX A-3

Instantiation of Classes

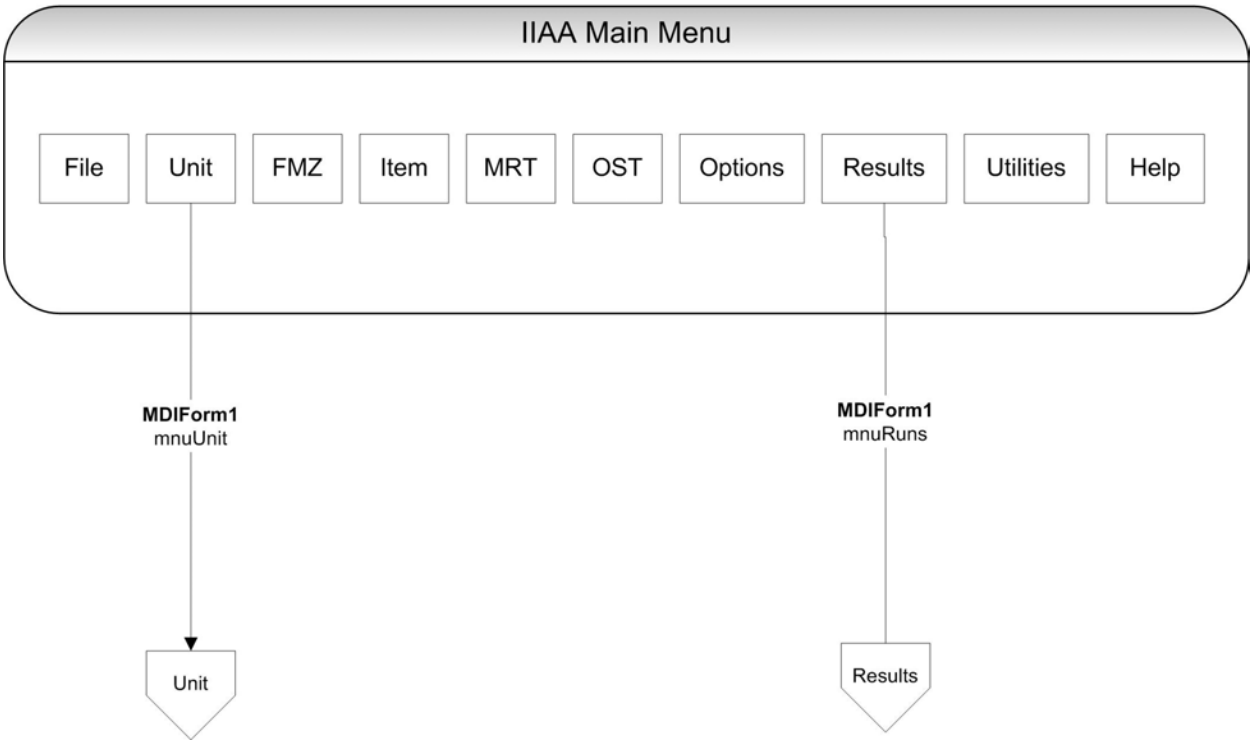
```
Classname: AACGraph
     Instantiated in Component AACrl At Line 12 As F
     Instantiated in Component AAC At Line 12 As F

Classname: EditFile
     Instantiated in Component MDIForm1 At Line 955 As F

Classname: EFT
     Instantiated in Component REPORT1 At Line 322 As aeftList

Classname: frmG1
     Instantiated in Component frmCNVC At Line 68 As frmNew

Classname: GridForm
     Instantiated in Component MDIForm1 At Line 196 As F
     Instantiated in Component MDIForm1 At Line 246 As F
     Instantiated in Component MDIForm1 At Line 270 As F
     Instantiated in Component MDIForm1 At Line 390 As F
     Instantiated in Component MDIForm1 At Line 642 As F
     Instantiated in Component MDIForm1 At Line 711 As F
     Instantiated in Component MDIForm1 At Line 764 As F
     Instantiated in Component MDIForm1 At Line 785 As F
     Instantiated in Component MDIForm1 At Line 1062 As F
     Instantiated in Component MDIForm1 At Line 1107 As F
     Instantiated in Component MDIForm1 At Line 1240 As F

Classname: View32
     Instantiated in Component B3Commonia At Line 205 As frmView
     Instantiated in Component B3Commonia At Line 329 As frmView
     Instantiated in Component B3Commonia At Line 595 As frmView
     Instantiated in Component MDIForm1 At Line 1285 As F
     Instantiated in Component frmOptSm At Line 170 As frmView
     Instantiated in Component frmSelectOptions At Line 286 As frmView
     Instantiated in Component dlgReports At Line 344 As frmView
     Instantiated in Component IIL At Line 236 As frmView
     Instantiated in Component frmXUMC At Line 68 As frmView
     Instantiated in Component frmCal2 At Line 341 As frmView
     Instantiated in Component LI32 At Line 729 As frmView
     Instantiated in Component EvalMO At Line 84 As frmView
     Instantiated in Component OptBust At Line 655 As frmView
     Instantiated in Component DARTSchk At Line 177 As frmView
```

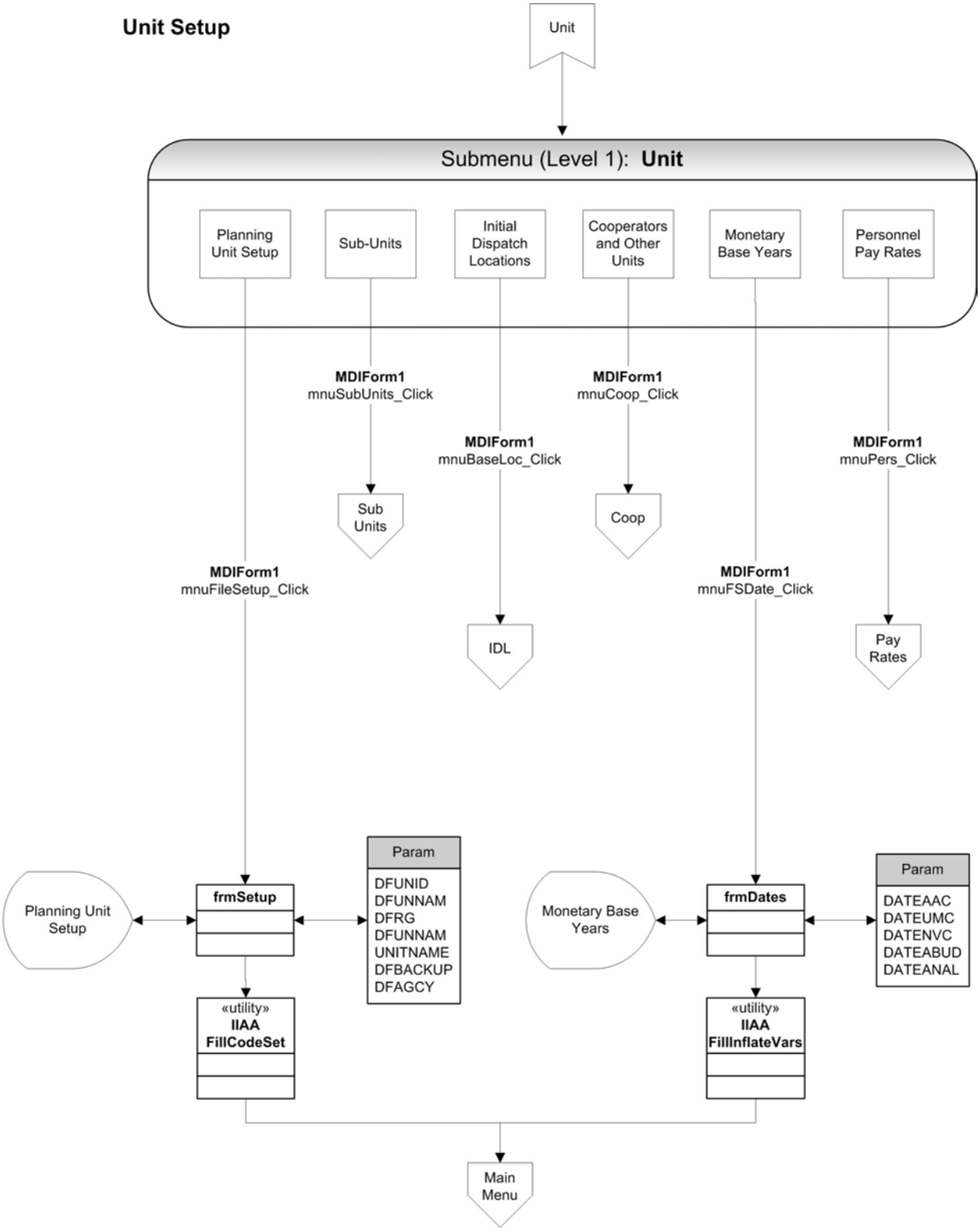
APPENDIX B

IIAA99 Program Logic Flow

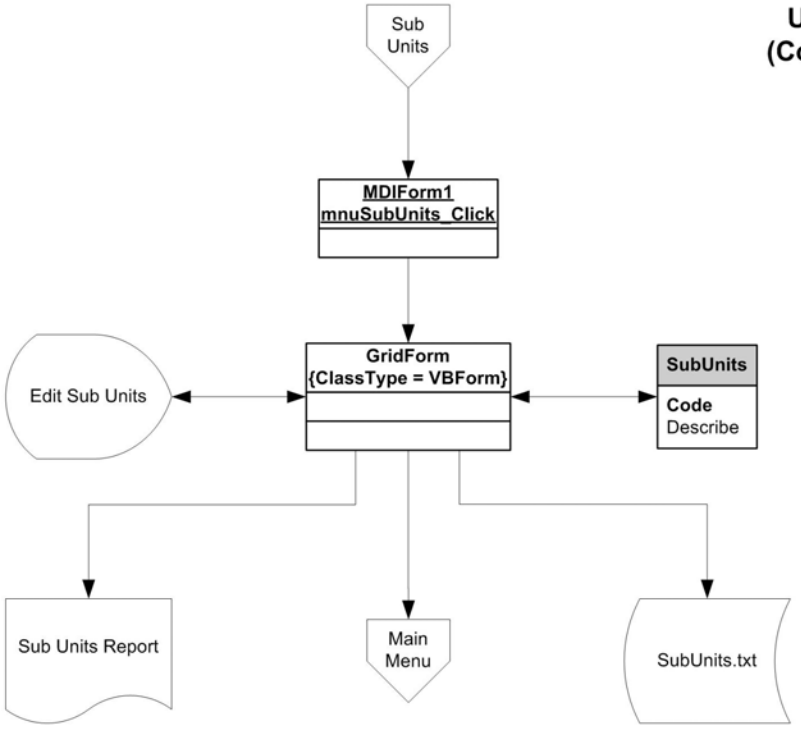

Program Initialization
Start
IIAA already running
Yes
Exit to Windows
No
Set up Directories and Paths
Registry
Key: IIAA99, Startup, Database
Disable all main menu items except those used to establish a path to the IIAA database.
Valid path to IIAA Database
No
MDIForm1 SetMenus
Yes
Open IIAA Database
Display the Main Menu
Set up codes appropriate to this Agency, update the IIAA database to the current revision level, assign QueryDefs to global variables, and calculate Inflation variables.
MDIForm1 InitMDB
"Reduced Availability" version of IIAA
Yes
Display warning message
Param
MSGAF
No
Main Menu

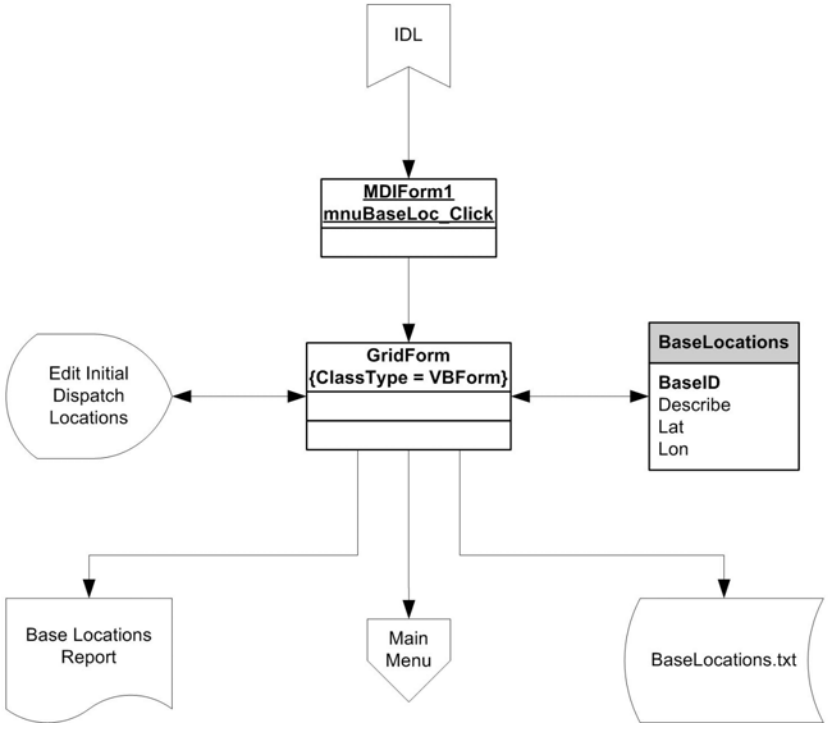
Main Menu


IIAA Main Menu
File
Unit
FMZ
Item
MRT
OST
Options
Results
Utilities
Help
MDIForm1
mnuUnit
MDIForm1
mnuRuns
Unit
Results

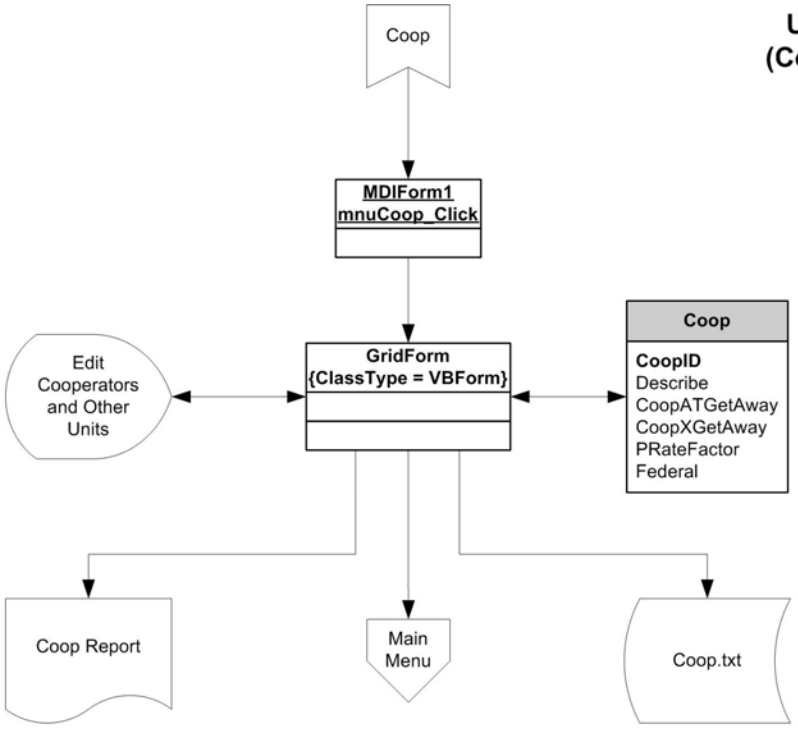

Unit Setup
Unit
Submenu (Level 1): Unit
Planning Unit Setup
Sub-Units
Initial Dispatch Locations
Cooperators and Other Units
Monetary Base Years
Personnel Pay Rates
MDIForm1
mnuSubUnits_Click
MDIForm1
mnuBaseLoc_Click
MDIForm1
mnuCoop_Click
MDIForm1
mnuPers_Click
Sub Units
Coop
MDIForm1
mnuFileSetup_Click
MDIForm1
mnuFSDate_Click
IDL
Pay Rates
Param
DFUNID
DFUNNAM
DFRG
DFUNNAM
UNITNAME
DFBACKUP
DFAGCY
Planning Unit Setup
frmSetup
«utility»
IIAA
FillCodeSet
Monetary Base Years
frmDates
«utility»
IIAA
FillInflateVars
Param
DATEAAC
DATEUMC
DATENVC
DATEABUD
DATEANAL
Main Menu

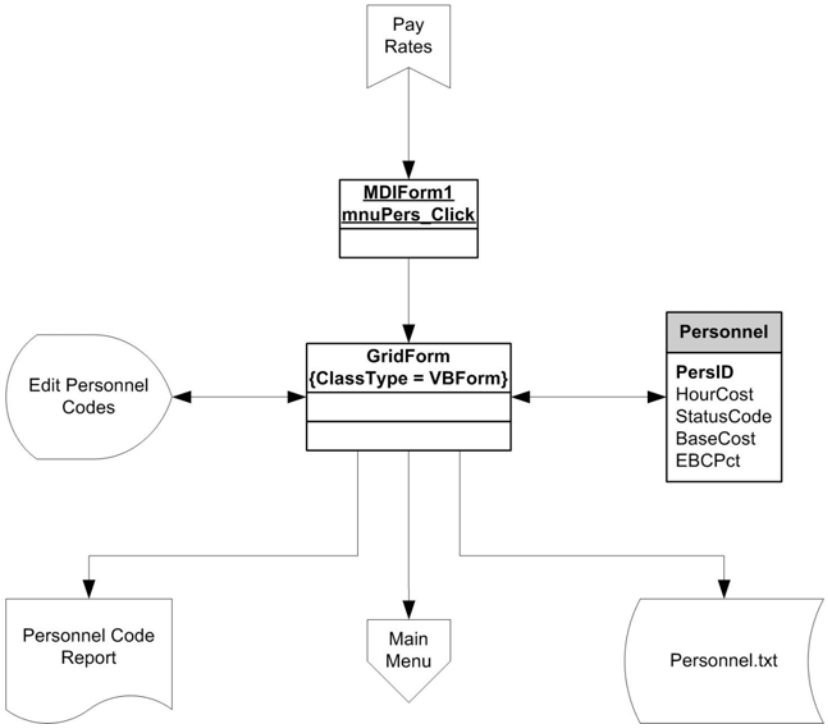
Unit Setup (Continued-1)

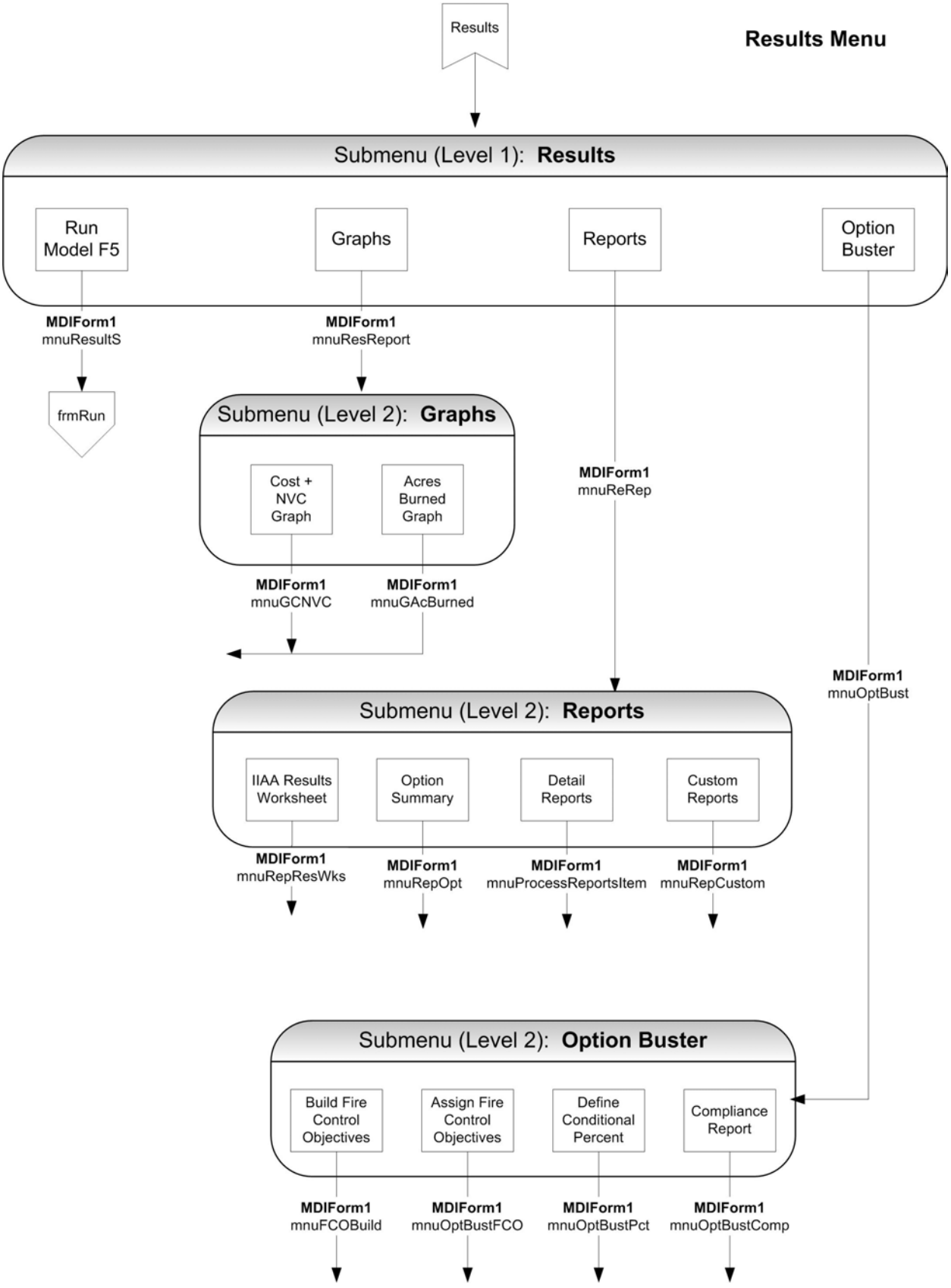


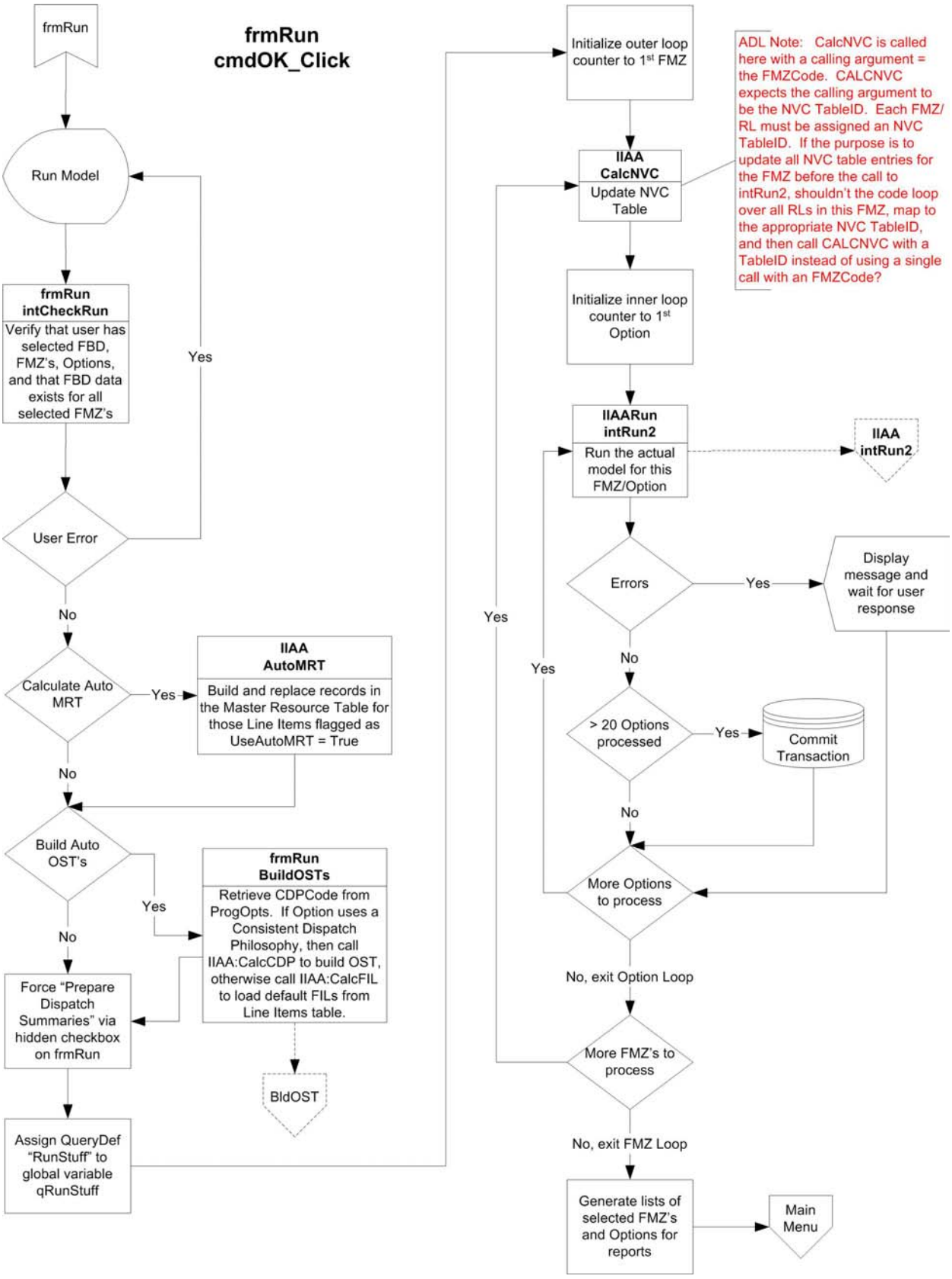


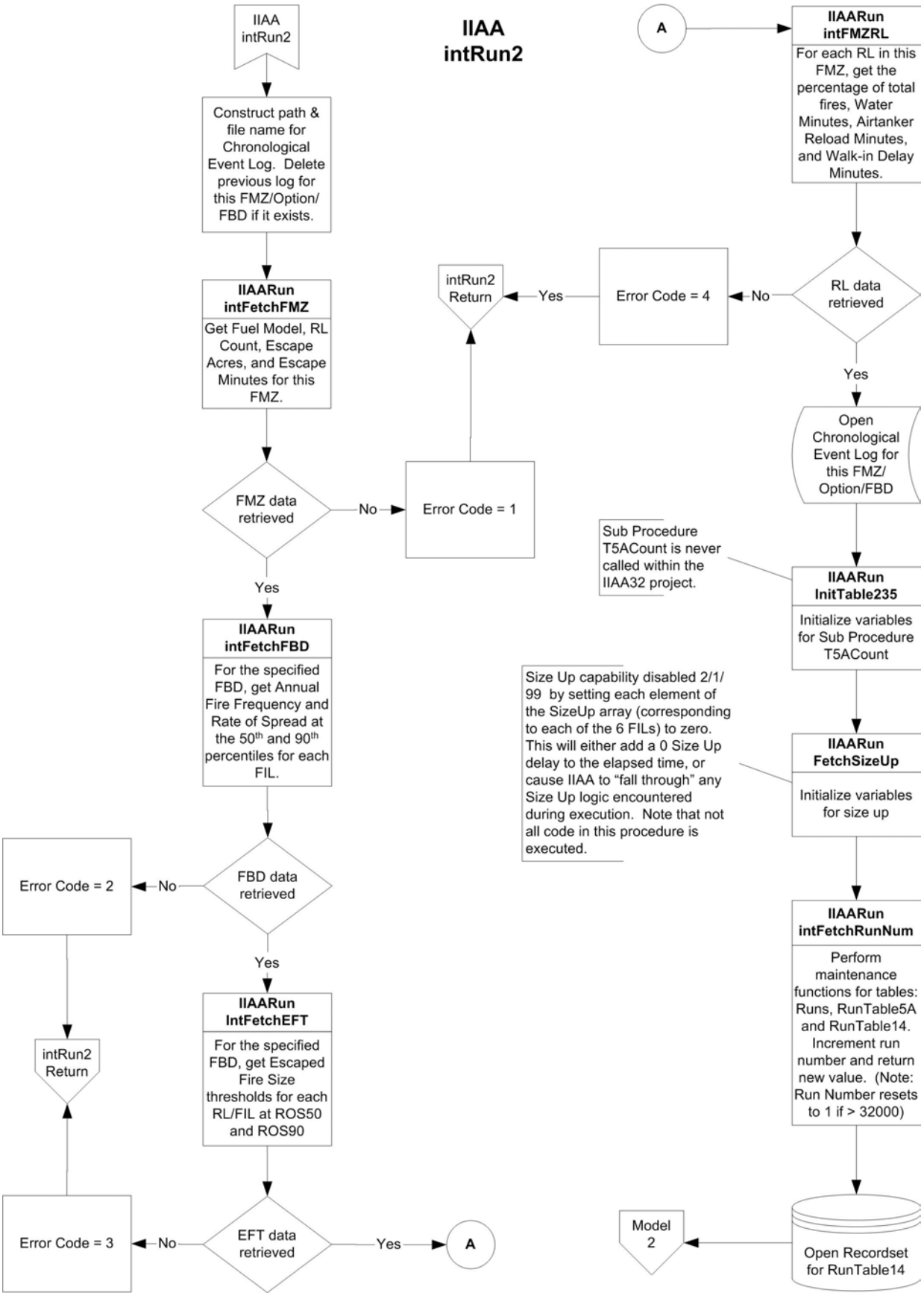
Unit Setup (Continued-2)

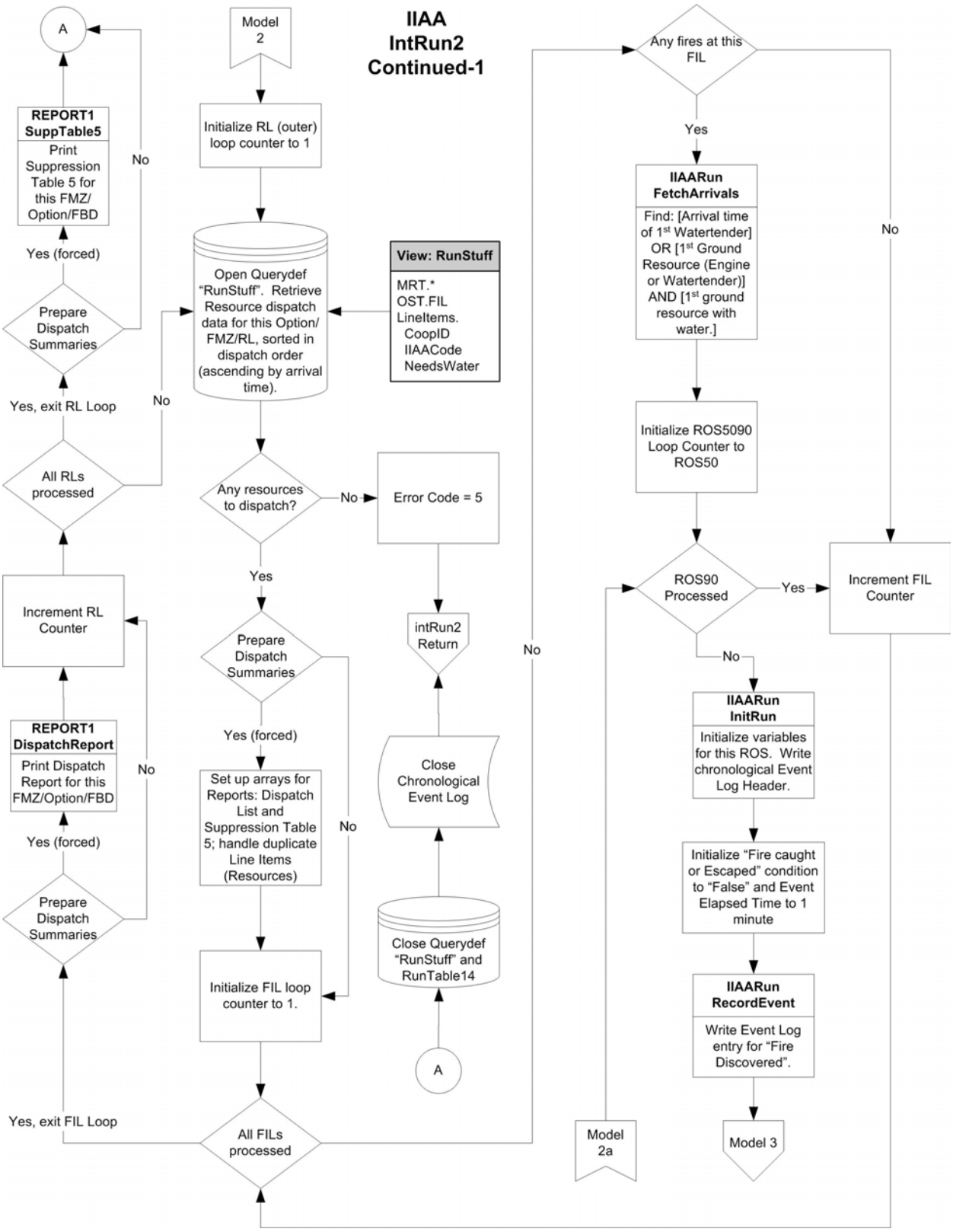


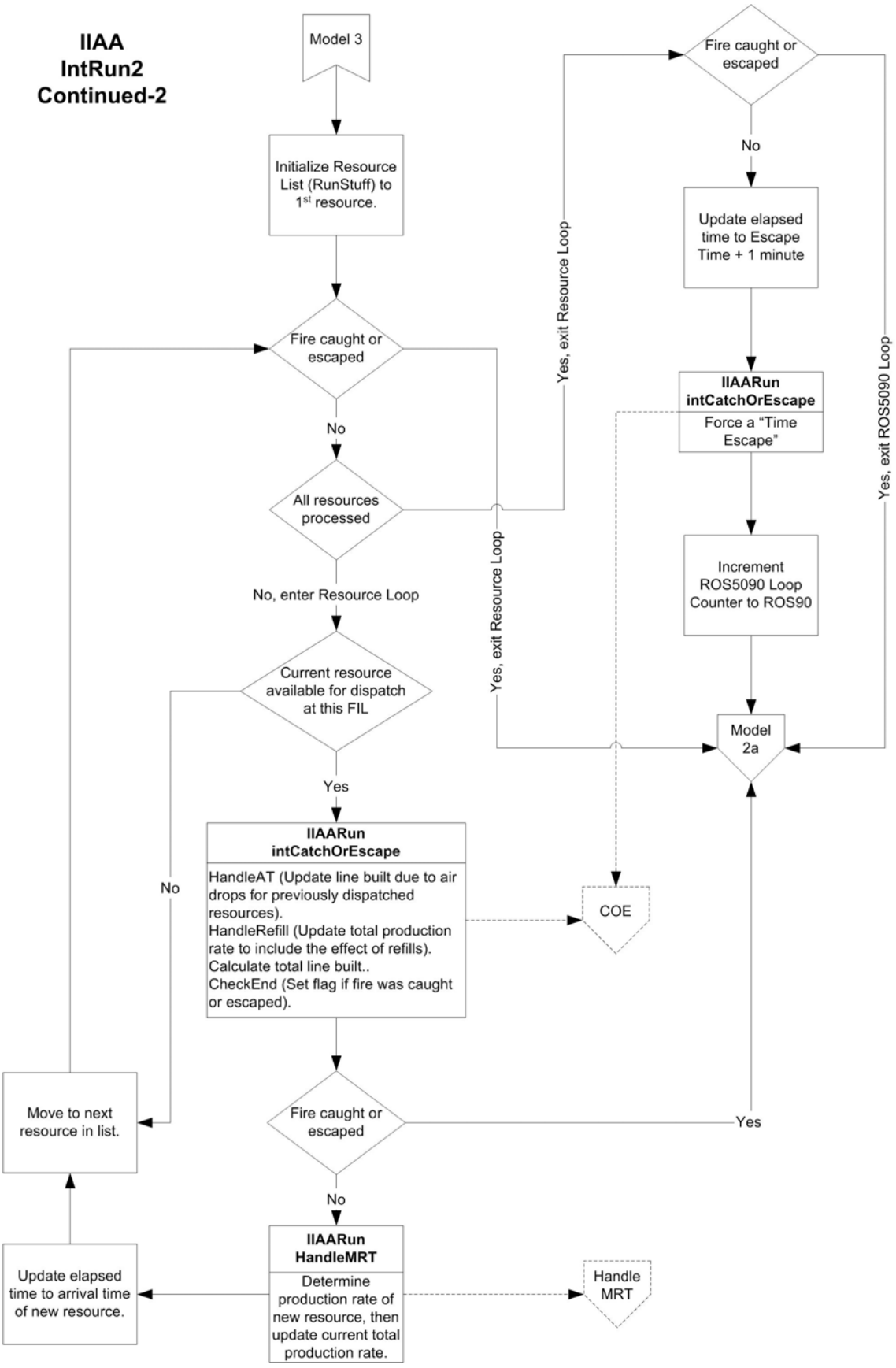


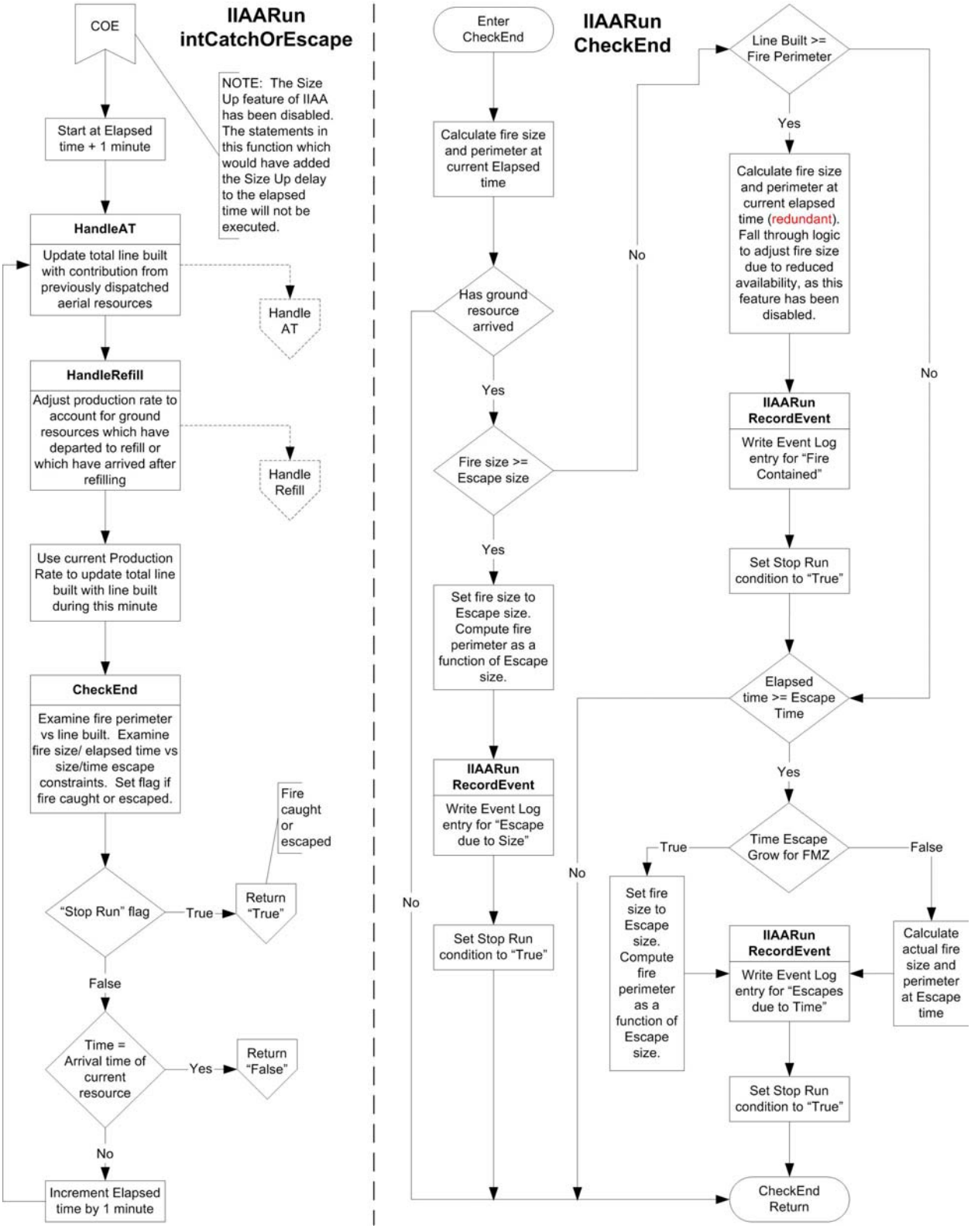

Results
Results Menu
Submenu (Level 1): Results
Run Model F5
Graphs
Reports
Option Buster
MDIForm1 mnuResultS
MDIForm1 mnuResReport
frmRun
Submenu (Level 2): Graphs
Cost + NVC Graph
Acres Burned Graph
MDIForm1 mnuGCNVC
MDIForm1 mnuGAcBurned
MDIForm1 mnuReRep
MDIForm1 mnuOptBust
Submenu (Level 2): Reports
IIAA Results Worksheet
Option Summary
Detail Reports
Custom Reports
MDIForm1 mnuRepResWks
MDIForm1 mnuRepOpt
MDIForm1 mnuProcessReportsItem
MDIForm1 mnuRepCustom
Submenu (Level 2): Option Buster
Build Fire Control Objectives
Assign Fire Control Objectives
Define Conditional Percent
Compliance Report
MDIForm1 mnuFCOBuild
MDIForm1 mnuOptBustFCO
MDIForm1 mnuOptBustPct
MDIForm1 mnuOptBustComp

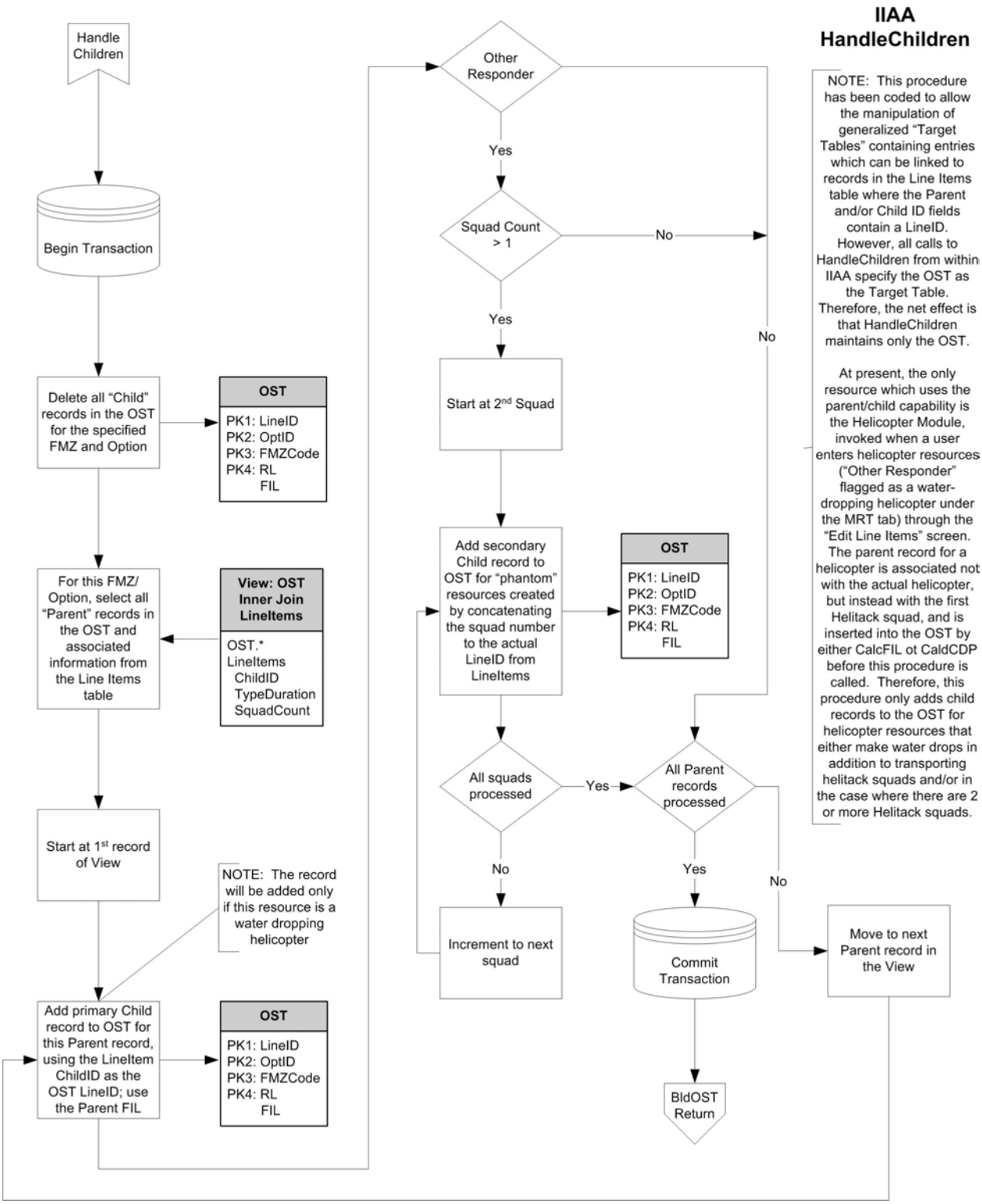

frmRun
frmRun
cmdOK_Click
Run Model
frmRun
intCheckRun
Verify that user has selected FBD, FMZ's, Options, and that FBD data exists for all selected FMZ's
Yes
User Error
No
Calculate Auto MRT
Yes
IIAA
AutoMRT
Build and replace records in the Master Resource Table for those Line Items flagged as UseAutoMRT = True
No
Build Auto OST's
Yes
frmRun
BuildOSTs
Retrieve CDPCode from ProgOpts. If Option uses a Consistent Dispatch Philosophy, then call IIAA:CalcCDP to build OST, otherwise call IIAA:CalcFIL to load default FILs from Line Items table.
No
Force "Prepare Dispatch Summaries" via hidden checkbox on frmRun
BldOST
Assign QueryDef "RunStuff" to global variable qRunStuff
Initialize outer loop counter to 1st FMZ
IIAA
CalcNVC
Update NVC Table
ADL Note: CalcNVC is called here with a calling argument = the FMZCode. CALCNVC expects the calling argument to be the NVC TableID. Each FMZ/RL must be assigned an NVC TableID. If the purpose is to update all NVC table entries for the FMZ before the call to intRun2, shouldn't the code loop over all RLs in this FMZ, map to the appropriate NVC TableID, and then call CALCNVC with a TableID instead of using a single call with an FMZCode?
Initialize inner loop counter to 1st Option
IIAARun
intRun2
Run the actual model for this FMZ/Option
IIAA
intRun2
Errors
Yes
Display message and wait for user response
Yes
No
> 20 Options processed
Yes
Commit Transaction
Yes
No
More Options to process
No, exit Option Loop
More FMZ's to process
Yes
No, exit FMZ Loop
Generate lists of selected FMZ's and Options for reports
Main Menu

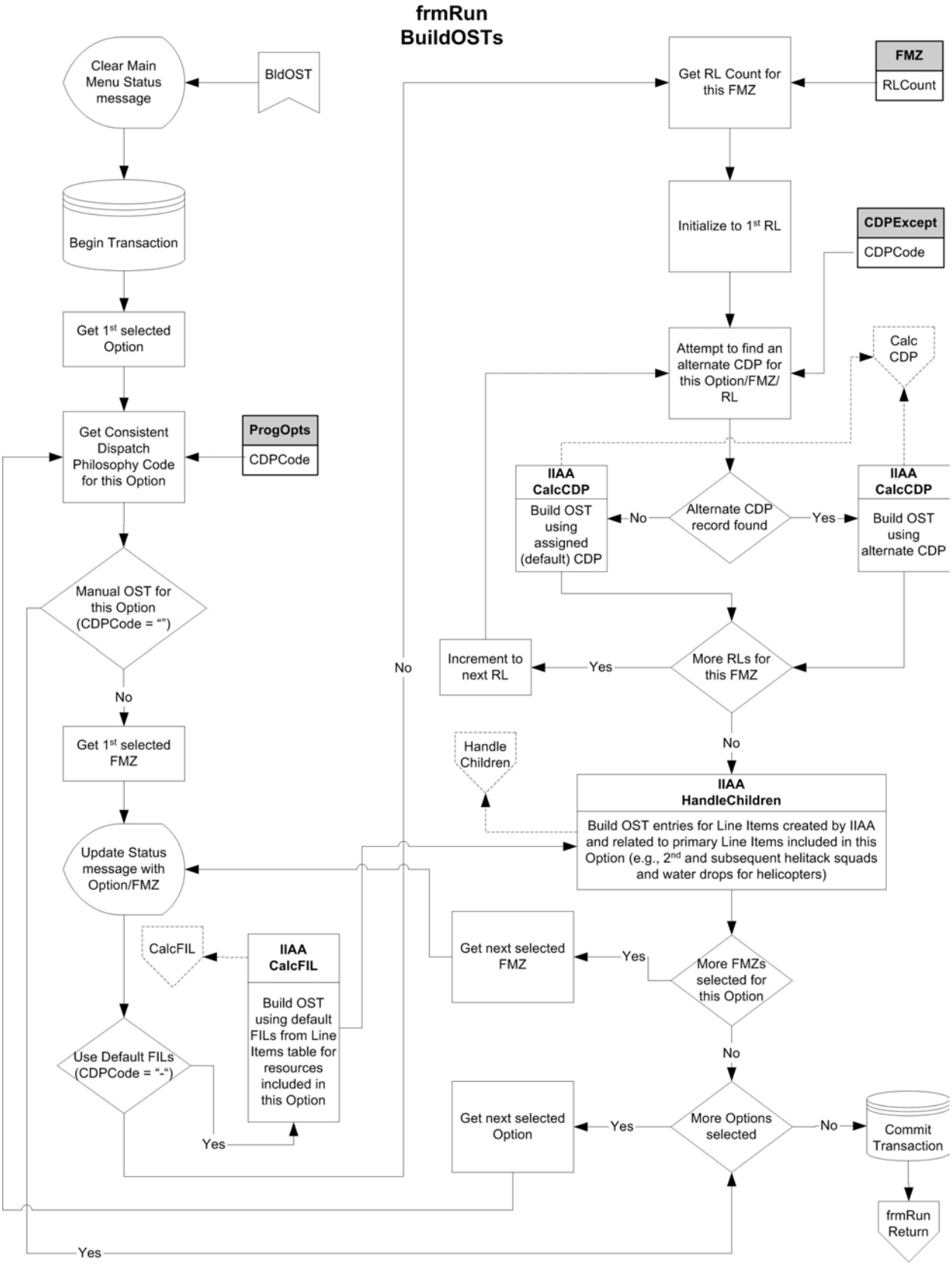

IIAA intRun2
IIAA intRun2
Construct path & file name for Chronological Event Log. Delete previous log for this FMZ/Option/ FBD if it exists.
IIAARun intFetchFMZ
Get Fuel Model, RL Count, Escape Acres, and Escape Minutes for this FMZ.
FMZ data retrieved
No
Error Code = 1
intRun2 Return
Yes
Error Code = 4
No
RL data retrieved
Yes
Open Chronological Event Log for this FMZ/ Option/FBD
IIAARun intFetchFBD
For the specified FBD, get Annual Fire Frequency and Rate of Spread at the 50th and 90th percentiles for each FIL.
FBD data retrieved
No
Error Code = 2
Yes
IIAARun IntFetchEFT
For the specified FBD, get Escaped Fire Size thresholds for each RL/FIL at ROS50 and ROS90
EFT data retrieved
No
Error Code = 3
Yes
A
intRun2 Return
A
IIAARun intFMZRL
For each RL in this FMZ, get the percentage of total fires, Water Minutes, Airtanker Reload Minutes, and Walk-in Delay Minutes.
Sub Procedure T5ACount is never called within the IIAA32 project.
IIAARun InitTable235
Initialize variables for Sub Procedure T5ACount
Size Up capability disabled 2/1/ 99 by setting each element of the SizeUp array (corresponding to each of the 6 FILs) to zero. This will either add a 0 Size Up delay to the elapsed time, or cause IIAA to “fall through” any Size Up logic encountered during execution. Note that not all code in this procedure is executed.
IIAARun FetchSizeUp
Initialize variables for size up
IIAARun intFetchRunNum
Perform maintenance functions for tables: Runs, RunTable5A and RunTable14. Increment run number and return new value. (Note: Run Number resets to 1 if > 32000)
Open Recordset for RunTable14
Model 2

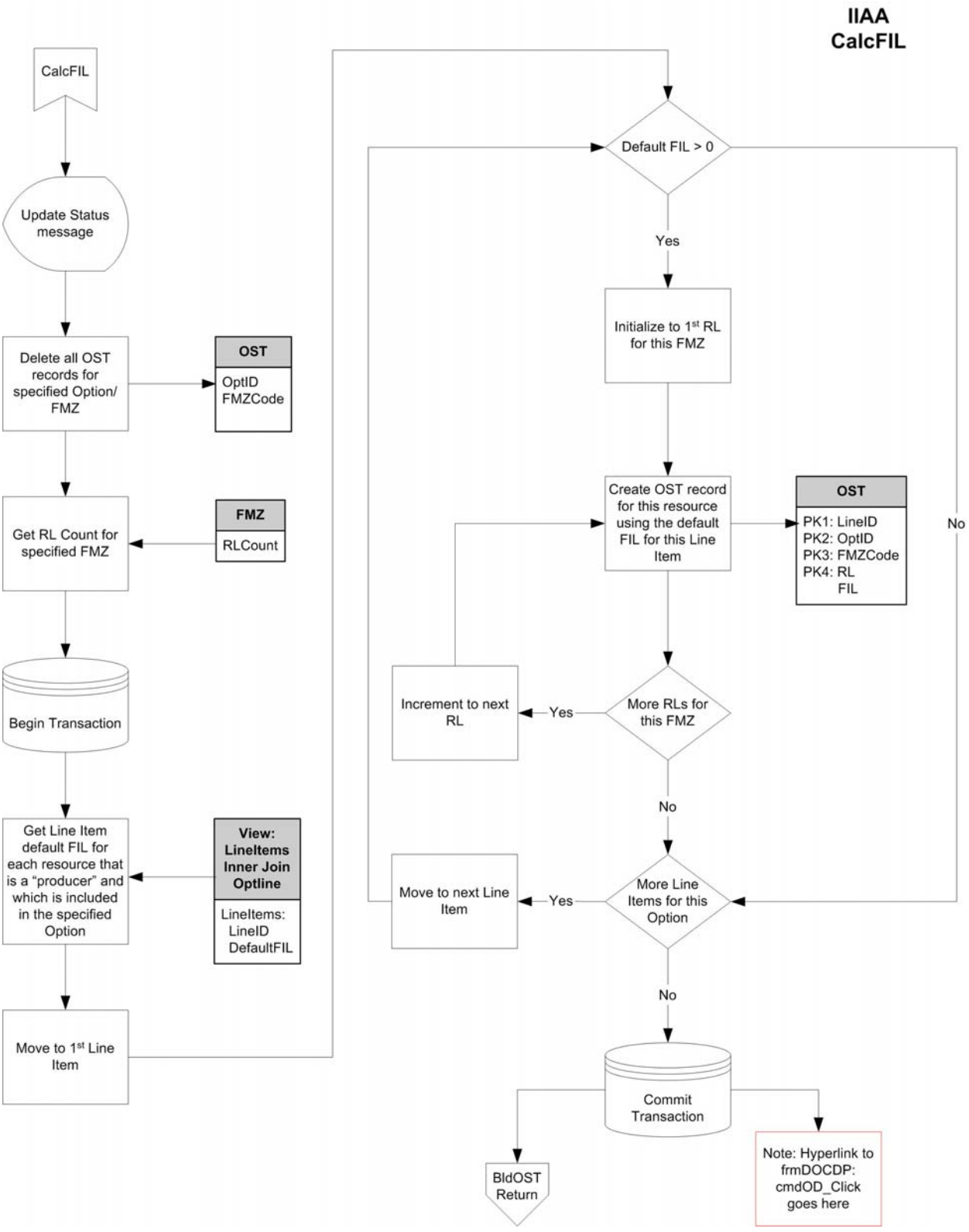

IIAA
IntRun2
Continued-1
A
Model 2
Initialize RL (outer) loop counter to 1
REPORT1 SuppTable5
Print Suppression Table 5 for this FMZ/ Option/FBD
No
Yes (forced)
Prepare Dispatch Summaries
Open Querydef "RunStuff". Retrieve Resource dispatch data for this Option/ FMZ/RL, sorted in dispatch order (ascending by arrival time).
View: RunStuff
MRT.*
OST.FIL
LineItems.
CoopID
IIAACode
NeedsWater
Yes, exit RL Loop
No
All RLs processed
Any resources to dispatch?
No
Error Code = 5
Yes
Increment RL Counter
Prepare Dispatch Summaries
intRun2 Return
REPORT1 DispatchReport
Print Dispatch Report for this FMZ/Option/FBD
No
Yes (forced)
Set up arrays for Reports: Dispatch List and Suppression Table 5; handle duplicate Line Items (Resources)
Close Chronological Event Log
Yes (forced)
Prepare Dispatch Summaries
No
Close Querydef "RunStuff" and RunTable14
Initialize FIL loop counter to 1.
A
Yes, exit FIL Loop
All FILs processed
No
Any fires at this FIL
Yes
No
IIAARun FetchArrivals
Find: [Arrival time of 1st Watertender] OR [1st Ground Resource (Engine or Watertender)] AND [1st ground resource with water.]
Initialize ROS5090 Loop Counter to ROS50
ROS90 Processed
Yes
Increment FIL Counter
No
IIAARun InitRun
Initialize variables for this ROS. Write chronological Event Log Header.
Initialize "Fire caught or Escaped" condition to "False" and Event Elapsed Time to 1 minute
IIAARun RecordEvent
Write Event Log entry for "Fire Discovered".
Model 2a
Model 3

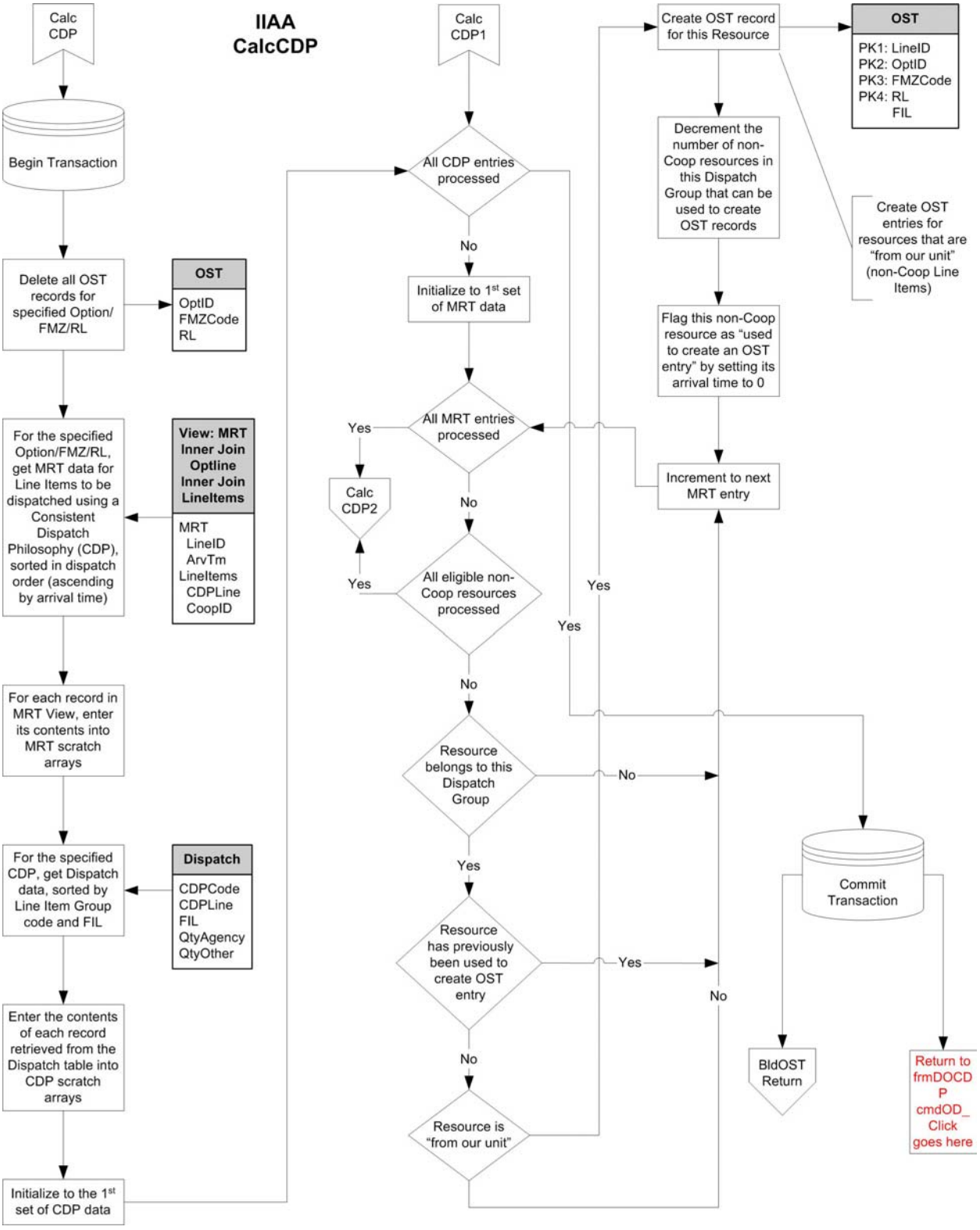

IIAA
IntRun2
Continued-2
Model 3
Initialize Resource List (RunStuff) to 1st resource.
Fire caught or escaped
No
All resources processed
No, enter Resource Loop
Current resource available for dispatch at this FIL
Yes
IIAARun intCatchOrEscape
HandleAT (Update line built due to air drops for previously dispatched resources).
HandleRefill (Update total production rate to include the effect of refills).
Calculate total line built..
CheckEnd (Set flag if fire was caught or escaped).
COE
No
Move to next resource in list.
Fire caught or escaped
Yes
No
IIAARun HandleMRT
Determine production rate of new resource, then update current total production rate.
Handle MRT
Update elapsed time to arrival time of new resource.
Yes, exit Resource Loop
Fire caught or escaped
No
Update elapsed time to Escape Time + 1 minute
IIAARun intCatchOrEscape
Force a "Time Escape"
Increment ROS5090 Loop Counter to ROS90
Yes, exit Resource Loop
Yes, exit ROS5090 Loop
Model 2a

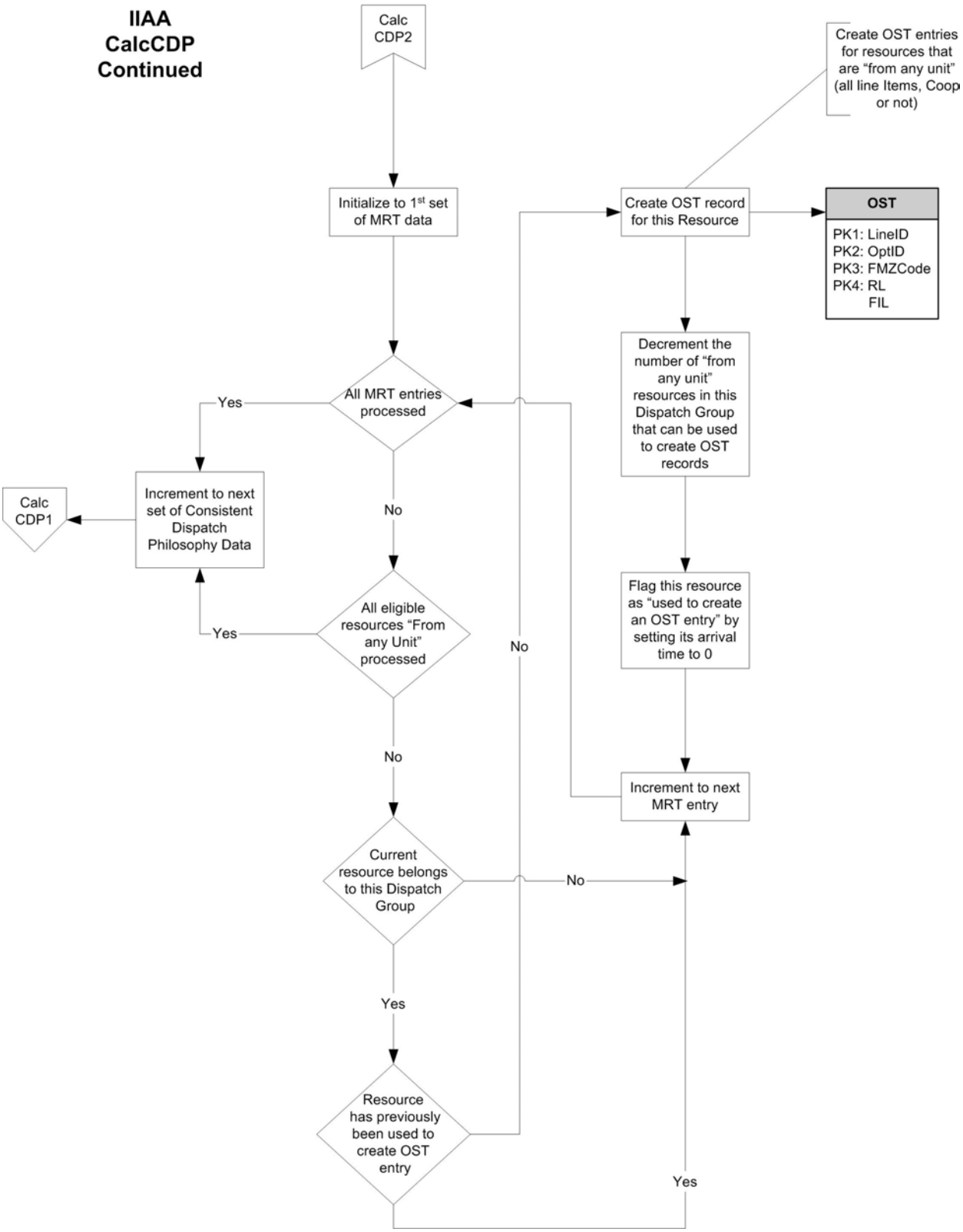

IIAARun
intCatchOrEscape
COE
NOTE: The Size Up feature of IIAA has been disabled. The statements in this function which would have added the Size Up delay to the elapsed time will not be executed.
Start at Elapsed time + 1 minute
HandleAT
Update total line built with contribution from previously dispatched aerial resources
Handle AT
HandleRefill
Adjust production rate to account for ground resources which have departed to refill or which have arrived after refilling
Handle Refill
Use current Production Rate to update total line built with line built during this minute
CheckEnd
Examine fire perimeter vs line built. Examine fire size/ elapsed time vs size/time escape constraints. Set flag if fire caught or escaped.
Fire caught or escaped
"Stop Run" flag
True
Return "True"
False
Time = Arrival time of current resource
Yes
Return "False"
No
Increment Elapsed time by 1 minute
IIAARun
CheckEnd
Enter CheckEnd
Calculate fire size and perimeter at current Elapsed time
Has ground resource arrived
Yes
No
Fire size >= Escape size
Yes
No
Set fire size to Escape size. Compute fire perimeter as a function of Escape size.
IIAARun RecordEvent
Write Event Log entry for "Escape due to Size"
Set Stop Run condition to "True"
Line Built >= Fire Perimeter
Yes
No
Calculate fire size and perimeter at current elapsed time (redundant). Fall through logic to adjust fire size due to reduced availability, as this feature has been disabled.
IIAARun RecordEvent
Write Event Log entry for "Fire Contained"
Set Stop Run condition to "True"
Elapsed time >= Escape Time
Yes
No
Time Escape Grow for FMZ
True
False
Set fire size to Escape size. Compute fire perimeter as a function of Escape size.
Calculate actual fire size and perimeter at Escape time
IIAARun RecordEvent
Write Event Log entry for "Escapes due to Time"
Set Stop Run condition to "True"
CheckEnd Return

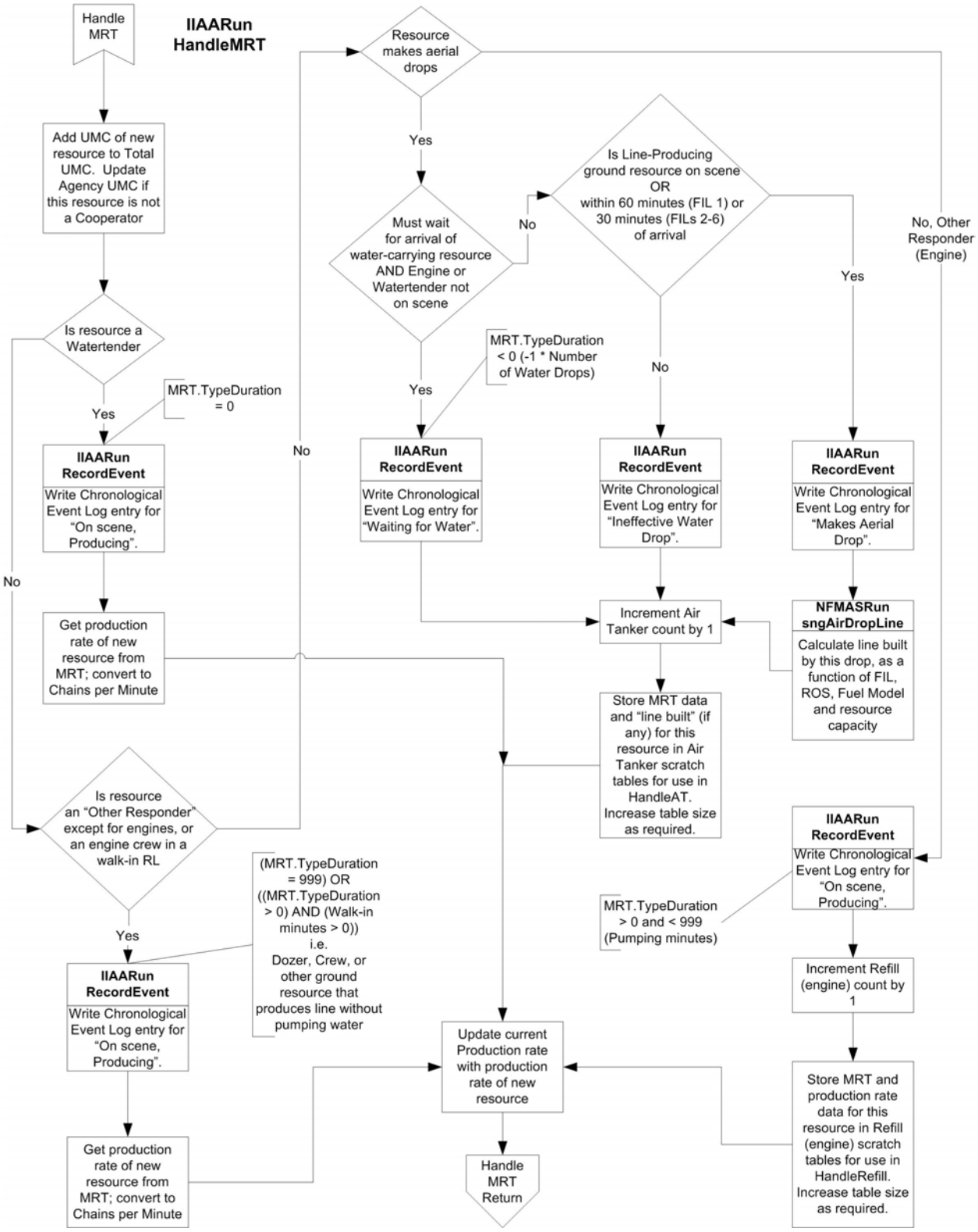

IIAA
HandleChildren
Handle Children
Begin Transaction
Delete all "Child" records in the OST for the specified FMZ and Option
OST
PK1: LineID
PK2: OptID
PK3: FMZCode
PK4: RL
FIL
For this FMZ/ Option, select all "Parent" records in the OST and associated information from the Line Items table
View: OST Inner Join LineItems
OST.*
LineItems
ChildID
TypeDuration
SquadCount
Start at 1st record of View
NOTE: The record will be added only if this resource is a water dropping helicopter
Add primary Child record to OST for this Parent record, using the LineItem ChildID as the OST LineID; use the Parent FIL
OST
PK1: LineID
PK2: OptID
PK3: FMZCode
PK4: RL
FIL
Other Responder
Yes
Squad Count > 1
No
Yes
Start at 2nd Squad
Add secondary Child record to OST for "phantom" resources created by concatenating the squad number to the actual LineID from LineItems
OST
PK1: LineID
PK2: OptID
PK3: FMZCode
PK4: RL
FIL
All squads processed
Yes
All Parent records processed
No
Yes
No
Increment to next squad
Commit Transaction
Move to next Parent record in the View
BldOST Return
No
NOTE: This procedure has been coded to allow the manipulation of generalized "Target Tables" containing entries which can be linked to records in the Line Items table where the Parent and/or Child ID fields contain a LineID. However, all calls to HandleChildren from within IIAA specify the OST as the Target Table. Therefore, the net effect is that HandleChildren maintains only the OST.
At present, the only resource which uses the parent/child capability is the Helicopter Module, invoked when a user enters helicopter resources ("Other Responder" flagged as a water-dropping helicopter under the MRT tab) through the "Edit Line Items" screen. The parent record for a helicopter is associated not with the actual helicopter, but instead with the first Helitack squad, and is inserted into the OST by either CalcFIL ot CaldCDP before this procedure is called. Therefore, this procedure only adds child records to the OST for helicopter resources that either make water drops in addition to transporting helitack squads and/or in the case where there are 2 or more Helitack squads.

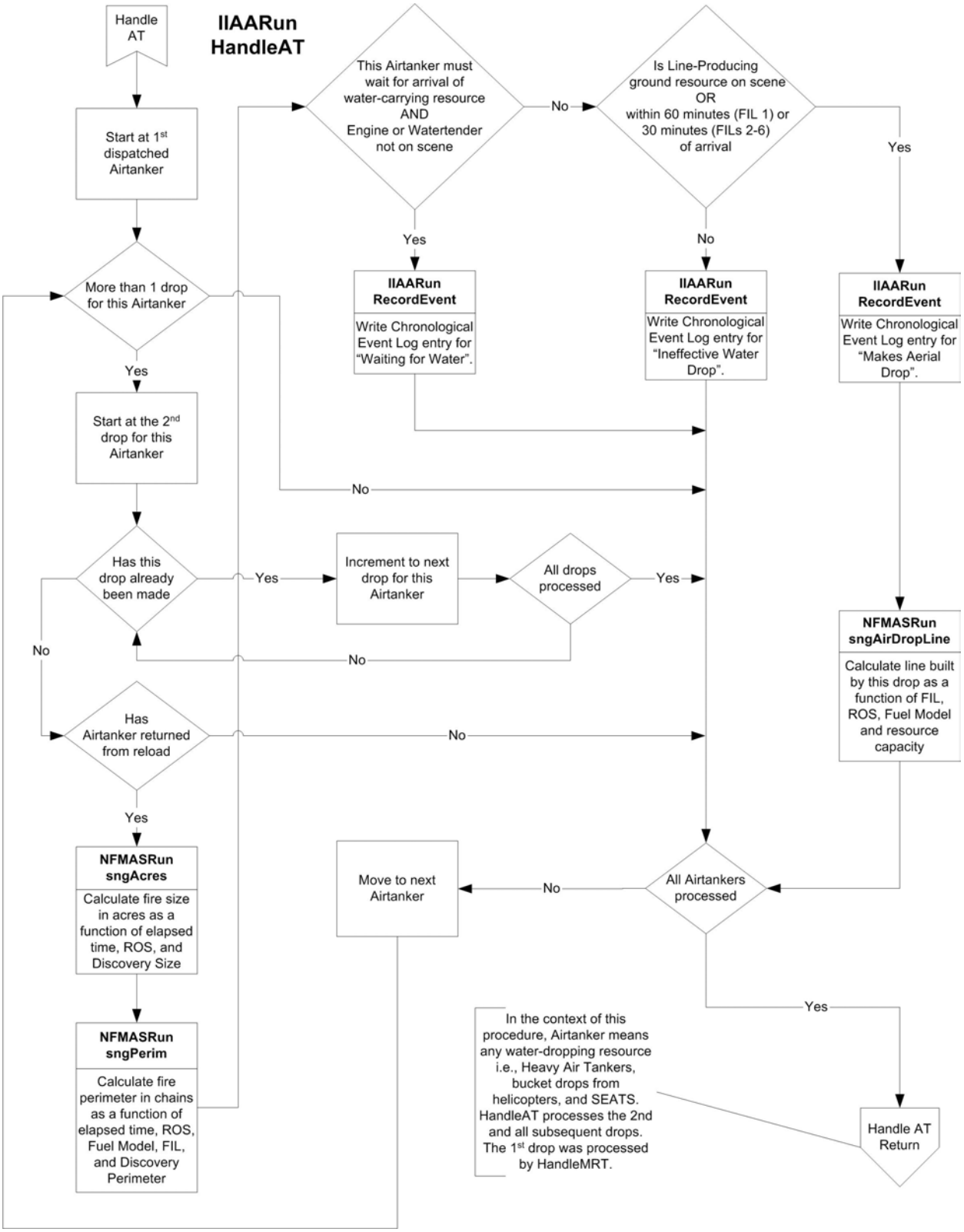

frmRun
BuildOSTs
Clear Main Menu Status message
BldOST
Begin Transaction
Get 1st selected Option
Get Consistent Dispatch Philosophy Code for this Option
ProgOpts
CDPCode
Manual OST for this Option (CDPCode = "")
No
Get 1st selected FMZ
Update Status message with Option/FMZ
CalcFIL
IIAA CalcFIL
Build OST using default FILs from Line Items table for resources included in this Option
Use Default FILs (CDPCode = "-")
Yes
Yes
Get RL Count for this FMZ
FMZ
RLCount
Initialize to 1st RL
CDPExcept
CDPCode
Attempt to find an alternate CDP for this Option/FMZ/ RL
Calc CDP
IIAA CalcCDP
Build OST using assigned (default) CDP
No
Alternate CDP record found
Yes
IIAA CalcCDP
Build OST using alternate CDP
No
Increment to next RL
Yes
More RLs for this FMZ
No
Handle Children
IIAA HandleChildren
Build OST entries for Line Items created by IIAA and related to primary Line Items included in this Option (e.g., 2nd and subsequent helitack squads and water drops for helicopters)
Get next selected FMZ
Yes
More FMZs selected for this Option
No
Get next selected Option
Yes
More Options selected
No
Commit Transaction
frmRun Return
Yes

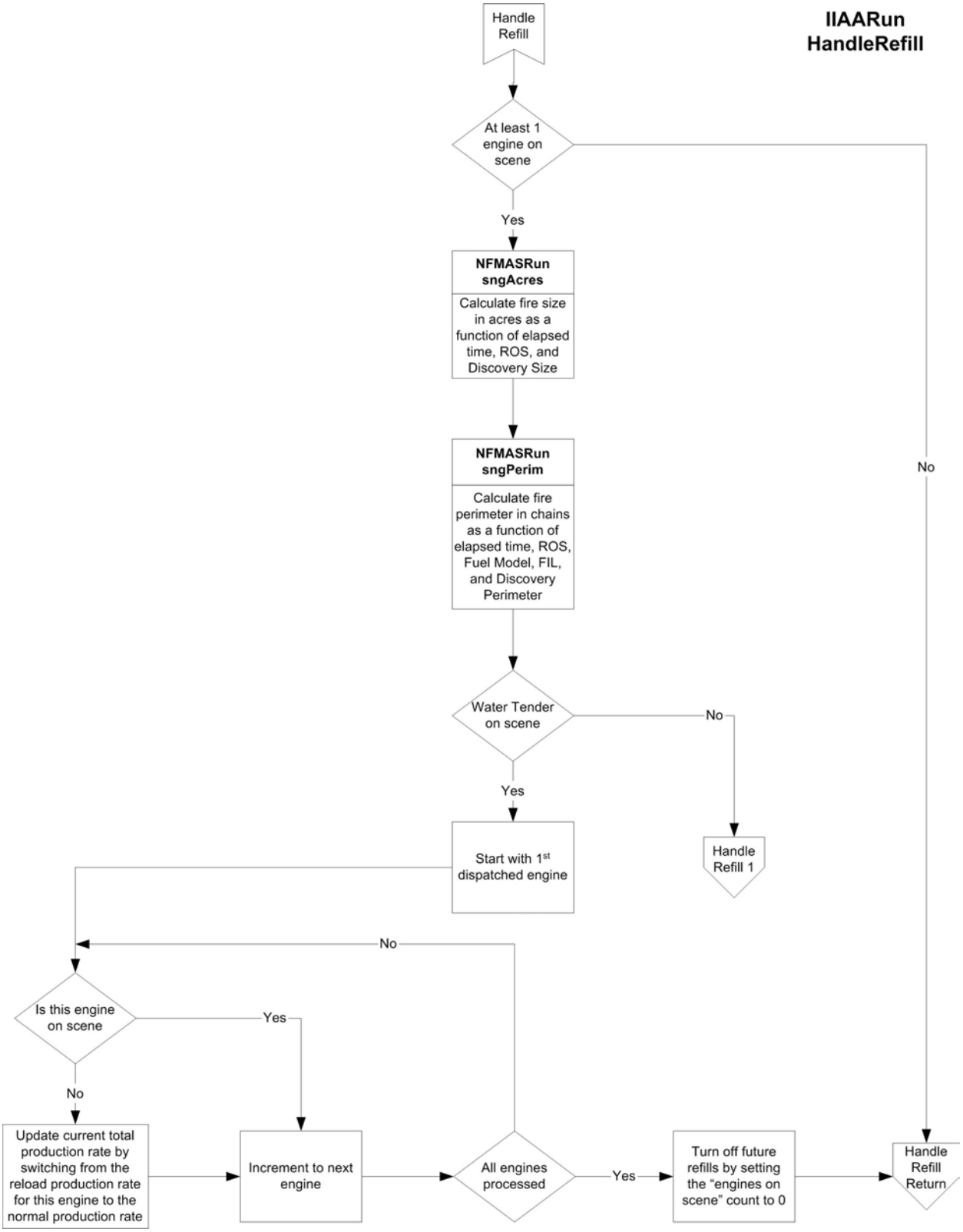

IIAA
CalcFIL
CalcFIL
Update Status message
Delete all OST records for specified Option/ FMZ
OST
OptID
FMZCode
Get RL Count for specified FMZ
FMZ
RLCount
Begin Transaction
Get Line Item default FIL for each resource that is a "producer" and which is included in the specified Option
View: LineItems Inner Join Optline
LineItems: LineID DefaultFIL
Move to 1st Line Item
Default FIL > 0
Yes
Initialize to 1st RL for this FMZ
Create OST record for this resource using the default FIL for this Line Item
OST
PK1: LineID
PK2: OptID
PK3: FMZCode
PK4: RL
FIL
No
Increment to next RL
Yes
More RLs for this FMZ
No
Move to next Line Item
Yes
More Line Items for this Option
No
Commit Transaction
BldOST Return
Note: Hyperlink to frmDOCDP: cmdOD_Click goes here


IIAA
CalcCDP
Calc CDP
Begin Transaction
Delete all OST records for specified Option/ FMZ/RL
OST
OptID
FMZCode
RL
For the specified Option/FMZ/RL, get MRT data for Line Items to be dispatched using a Consistent Dispatch Philosophy (CDP), sorted in dispatch order (ascending by arrival time)
View: MRT
Inner Join Optline
Inner Join LineItems
MRT
LineID
ArvTm
LineItems
CDPLine
CoopID
For each record in MRT View, enter its contents into MRT scratch arrays
For the specified CDP, get Dispatch data, sorted by Line Item Group code and FIL
Dispatch
CDPCode
CDPLine
FIL
QtyAgency
QtyOther
Enter the contents of each record retrieved from the Dispatch table into CDP scratch arrays
Initialize to the 1st set of CDP data
Calc CDP1
All CDP entries processed
No
Initialize to 1st set of MRT data
All MRT entries processed
Yes
Calc CDP2
No
All eligible non-Coop resources processed
Yes
No
Resource belongs to this Dispatch Group
No
Yes
Resource has previously been used to create OST entry
Yes
No
No
Resource is "from our unit"
Yes
Yes
Create OST record for this Resource
OST
PK1: LineID
PK2: OptID
PK3: FMZCode
PK4: RL
FIL
Decrement the number of non-Coop resources in this Dispatch Group that can be used to create OST records
Create OST entries for resources that are "from our unit" (non-Coop Line Items)
Flag this non-Coop resource as "used to create an OST entry" by setting its arrival time to 0
Increment to next MRT entry
Commit Transaction
BldOST Return
Return to frmDOCDP cmdOD_Click goes here

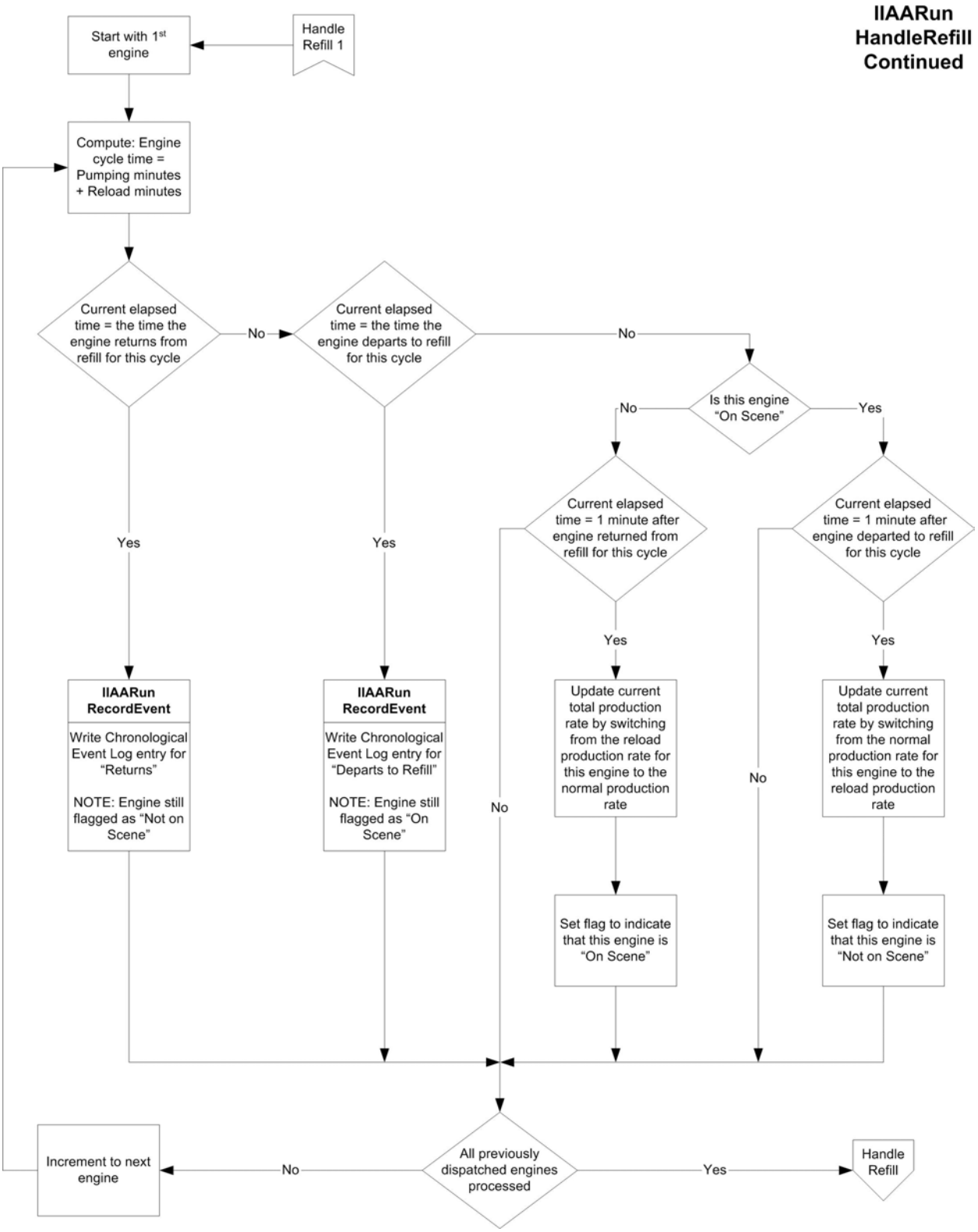

IIAA
CalcCDP
Continued
Calc CDP2
Initialize to 1st set of MRT data
All MRT entries processed
Yes
No
Increment to next set of Consistent Dispatch Philosophy Data
Calc CDP1
All eligible resources "From any Unit" processed
Yes
No
Current resource belongs to this Dispatch Group
No
Yes
Resource has previously been used to create OST entry
No
Yes
Create OST record for this Resource
Create OST entries for resources that are "from any unit" (all line Items, Coop or not)
OST
PK1: LineID
PK2: OptID
PK3: FMZCode
PK4: RL
FIL
Decrement the number of "from any unit" resources in this Dispatch Group that can be used to create OST records
Flag this resource as "used to create an OST entry" by setting its arrival time to 0
Increment to next MRT entry


Handle MRT
IIAARun HandleMRT
Add UMC of new resource to Total UMC. Update Agency UMC if this resource is not a Cooperator
Is resource a Watertender
Yes
MRT.TypeDuration = 0
IIAARun RecordEvent
Write Chronological Event Log entry for "On scene, Producing".
No
Get production rate of new resource from MRT; convert to Chains per Minute
Is resource an "Other Responder" except for engines, or an engine crew in a walk-in RL
Yes
(MRT.TypeDuration = 999) OR ((MRT.TypeDuration > 0) AND (Walk-in minutes > 0)) i.e. Dozer, Crew, or other ground resource that produces line without pumping water
IIAARun RecordEvent
Write Chronological Event Log entry for "On scene, Producing".
Get production rate of new resource from MRT; convert to Chains per Minute
No
Resource makes aerial drops
Yes
Must wait for arrival of water-carrying resource AND Engine or Watertender not on scene
No
Is Line-Producing ground resource on scene OR within 60 minutes (FIL 1) or 30 minutes (FILs 2-6) of arrival
No
Yes
No, Other Responder (Engine)
MRT.TypeDuration < 0 (-1 * Number of Water Drops)
Yes
IIAARun RecordEvent
Write Chronological Event Log entry for "Waiting for Water".
IIAARun RecordEvent
Write Chronological Event Log entry for "Ineffective Water Drop".
IIAARun RecordEvent
Write Chronological Event Log entry for "Makes Aerial Drop".
Increment Air Tanker count by 1
NFMASRun sngAirDropLine
Calculate line built by this drop, as a function of FIL, ROS, Fuel Model and resource capacity
Store MRT data and "line built" (if any) for this resource in Air Tanker scratch tables for use in HandleAT. Increase table size as required.
IIAARun RecordEvent
Write Chronological Event Log entry for "On scene, Producing".
MRT.TypeDuration > 0 and < 999 (Pumping minutes)
Increment Refill (engine) count by 1
Store MRT and production rate data for this resource in Refill (engine) scratch tables for use in HandleRefill. Increase table size as required.
Update current Production rate with production rate of new resource
Handle MRT Return


Handle AT
IIAARun HandleAT
Start at 1st dispatched Airtanker
More than 1 drop for this Airtanker
Yes
Start at the 2nd drop for this Airtanker
Has this drop already been made
Yes
Increment to next drop for this Airtanker
All drops processed
Yes
No
No
Has Airtanker returned from reload
No
Yes
NFMASRun sngAcres
Calculate fire size in acres as a function of elapsed time, ROS, and Discovery Size
NFMASRun sngPerim
Calculate fire perimeter in chains as a function of elapsed time, ROS, Fuel Model, FIL, and Discovery Perimeter
This Airtanker must wait for arrival of water-carrying resource AND Engine or Watertender not on scene
No
Yes
IIAARun RecordEvent
Write Chronological Event Log entry for "Waiting for Water".
Is Line-Producing ground resource on scene OR within 60 minutes (FIL 1) or 30 minutes (FILs 2-6) of arrival
No
IIAARun RecordEvent
Write Chronological Event Log entry for "Ineffective Water Drop".
Yes
IIAARun RecordEvent
Write Chronological Event Log entry for "Makes Aerial Drop".
NFMASRun sngAirDropLine
Calculate line built by this drop as a function of FIL, ROS, Fuel Model and resource capacity
All Airtankers processed
No
Move to next Airtanker
Yes
In the context of this procedure, Airtanker means any water-dropping resource i.e., Heavy Air Tankers, bucket drops from helicopters, and SEATS. HandleAT processes the 2nd and all subsequent drops. The 1st drop was processed by HandleMRT.
Handle AT Return

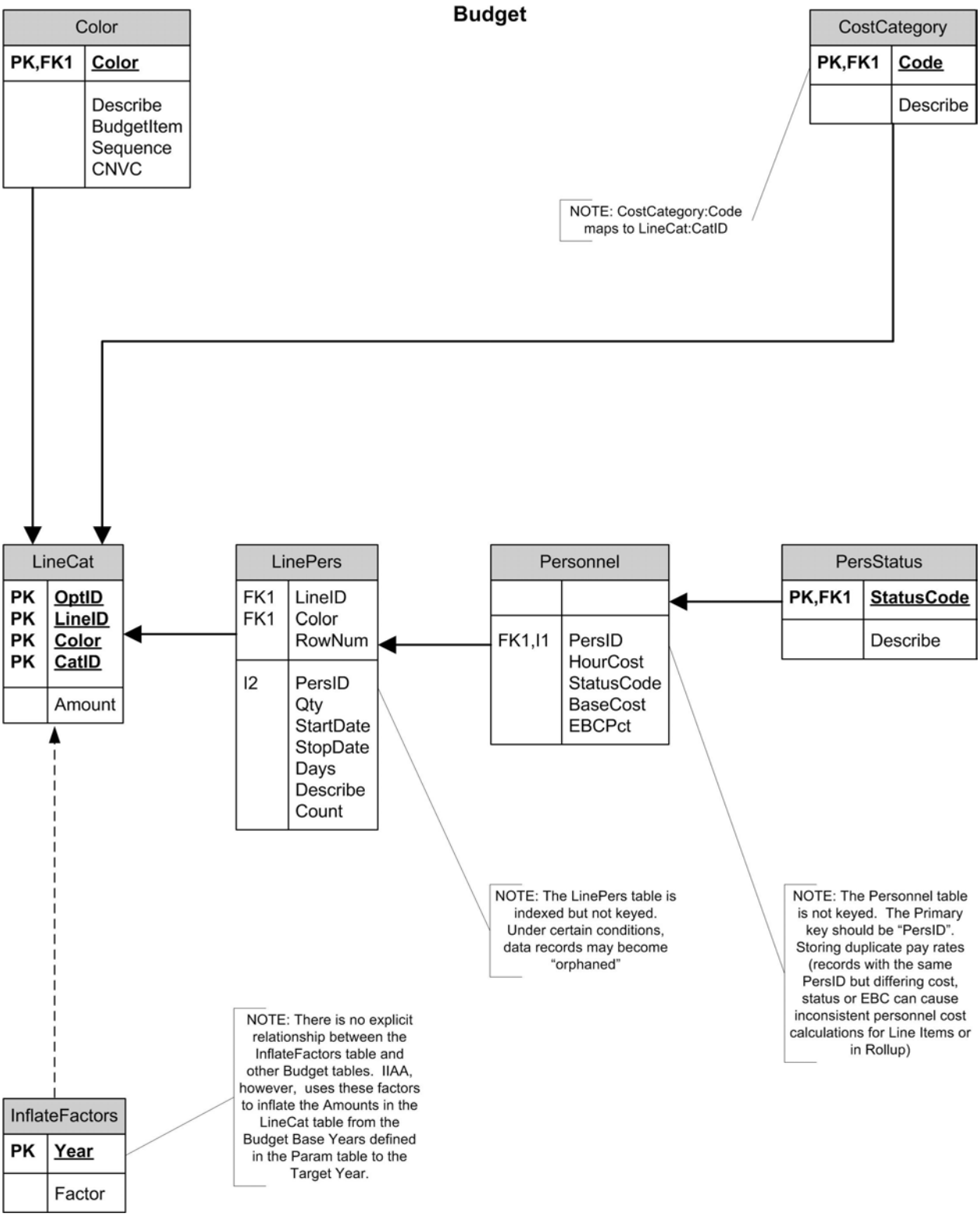

IIAARun
HandleRefill
Handle Refill
At least 1 engine on scene
Yes
No
NFMASRun sngAcres
Calculate fire size in acres as a function of elapsed time, ROS, and Discovery Size
NFMASRun sngPerim
Calculate fire perimeter in chains as a function of elapsed time, ROS, Fuel Model, FIL, and Discovery Perimeter
Water Tender on scene
No
Yes
Handle Refill 1
Start with 1st dispatched engine
No
Is this engine on scene
Yes
No
Update current total production rate by switching from the reload production rate for this engine to the normal production rate
Increment to next engine
All engines processed
Yes
Turn off future refills by setting the "engines on scene" count to 0
Handle Refill Return

IIAARun HandleRefill Continued

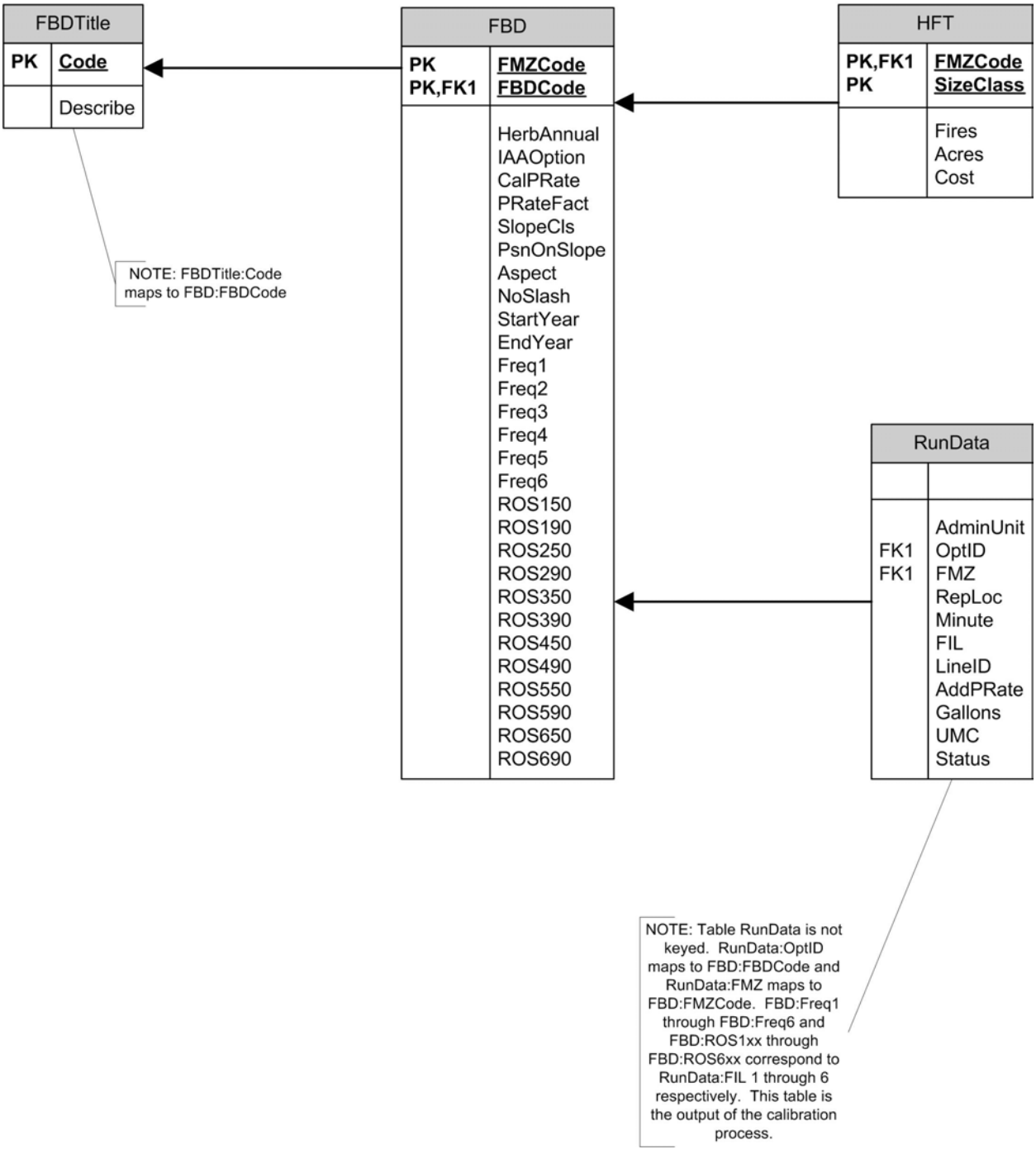


APPENDIX C

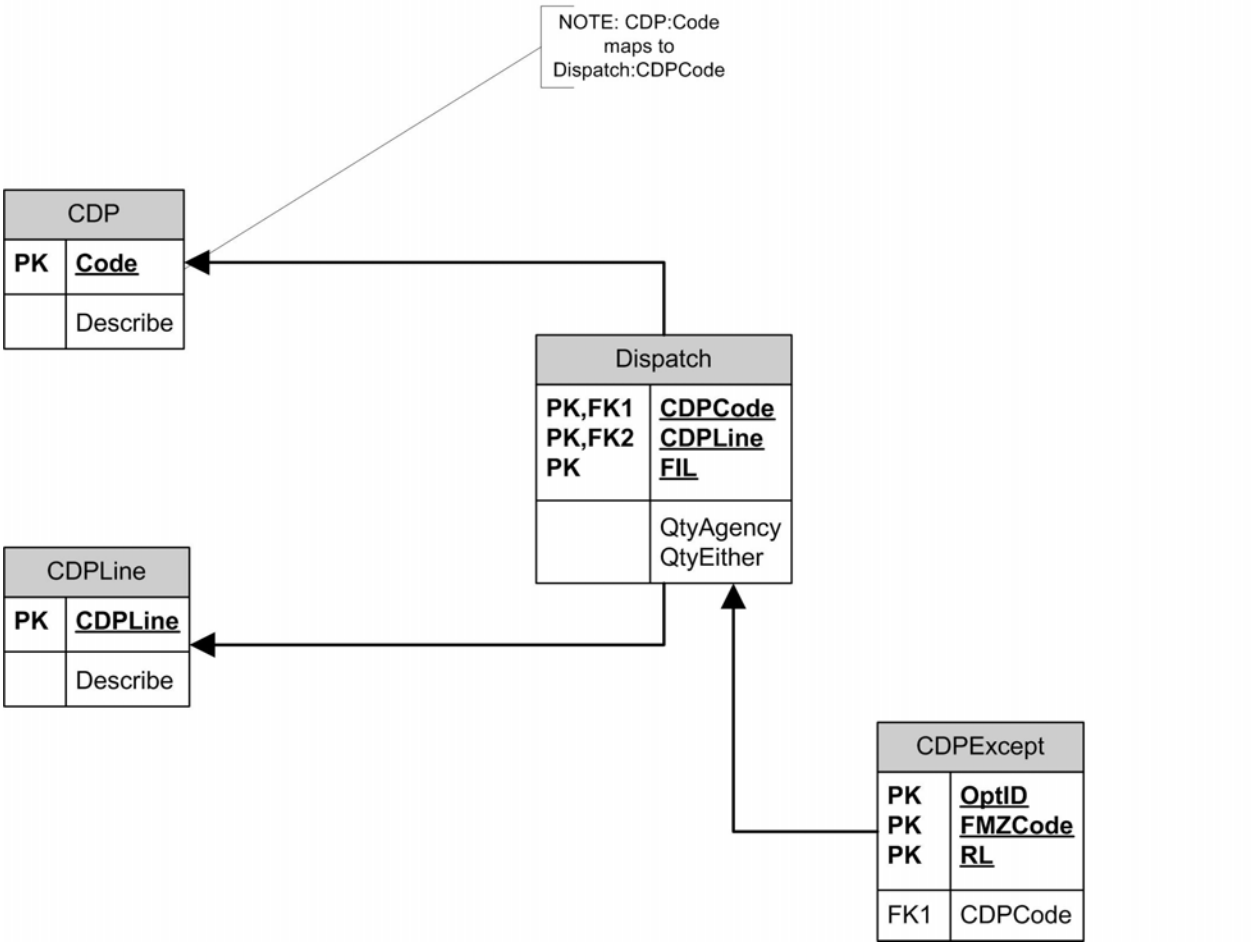
IIAA99 Database Relationships


Budget
Color
PK,FK1 Color
Describe
BudgetItem
Sequence
CNVC
CostCategory
PK,FK1 Code
Describe
NOTE: CostCategory:Code maps to LineCat:CatID
LineCat
PK OptID
PK LineID
PK Color
PK CatID
Amount
LinePers
FK1 LineID
FK1 Color
RowNum
I2 PersID
Qty
StartDate
StopDate
Days
Describe
Count
Personnel
FK1,I1 PersID
HourCost
StatusCode
BaseCost
EBCPct
PersStatus
PK,FK1 StatusCode
Describe
NOTE: The LinePers table is indexed but not keyed. Under certain conditions, data records may become "orphaned"
NOTE: The Personnel table is not keyed. The Primary key should be "PersID". Storing duplicate pay rates (records with the same PersID but differing cost, status or EBC can cause inconsistent personnel cost calculations for Line Items or in Rollup)
InflateFactors
PK Year
Factor
NOTE: There is no explicit relationship between the InflateFactors table and other Budget tables. IIAA, however, uses these factors to inflate the Amounts in the LineCat table from the Budget Base Years defined in the Param table to the Target Year.

Calibration

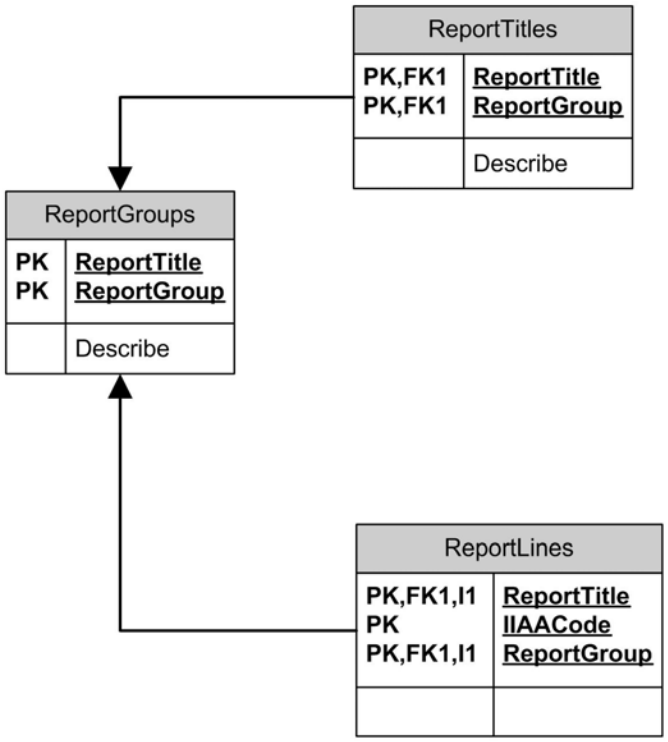


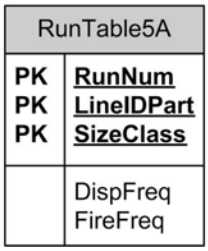
Consistent Dispatch Philosophy

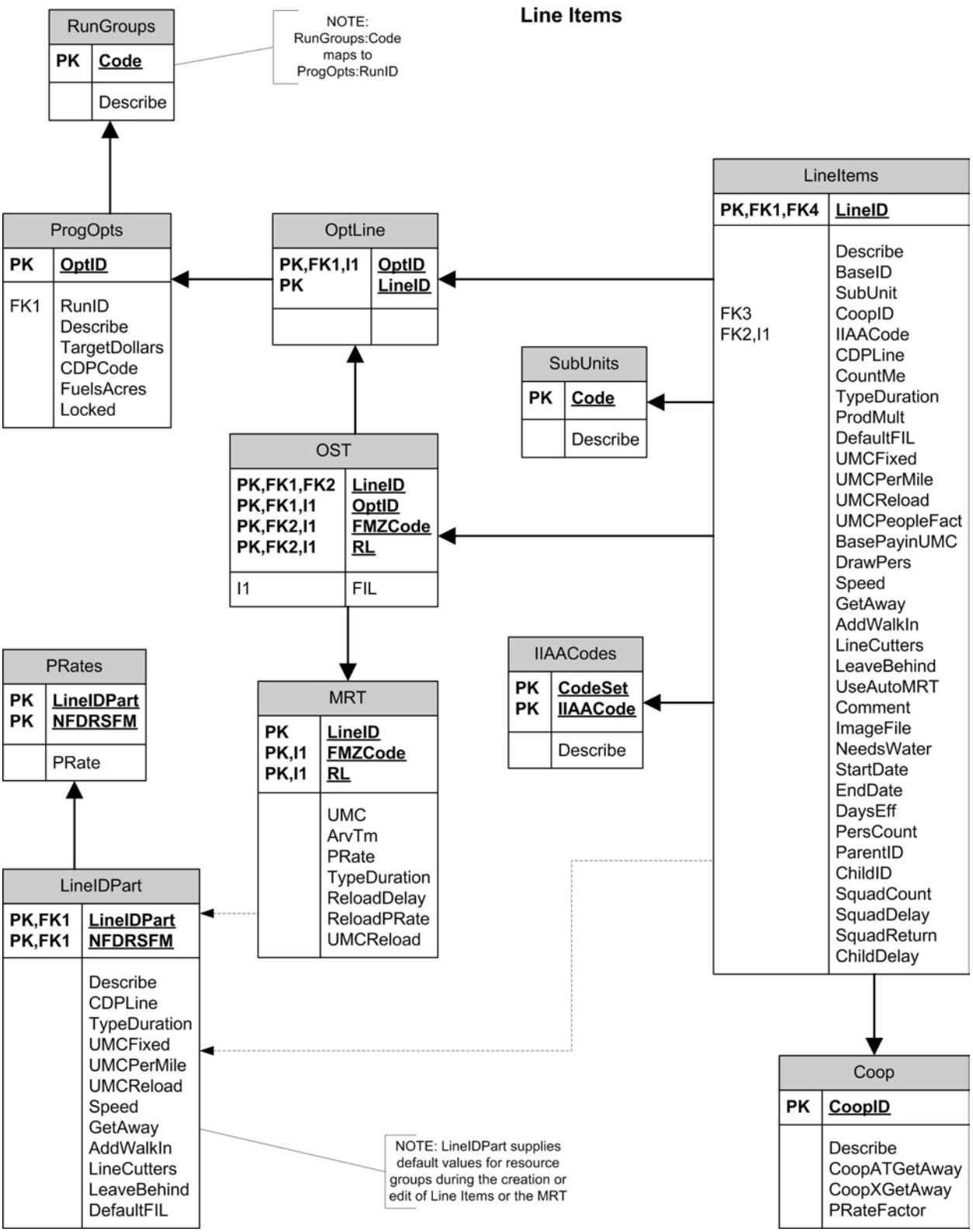


Dead Tables

NOTE: These tables are referenced in the IIAA source code, but serve no current useful purpose in the system. They cannot be arbitrarily removed from the database without corresponding modifications to the IIAA source.






Line Items
RunGroups
PK Code
Describe
NOTE: RunGroups:Code maps to ProgOpts:RunID
ProgOpts
PK OptID
FK1 RunID
Describe
TargetDollars
CDPCode
FuelsAcres
Locked
OptLine
PK,FK1,I1 OptID
PK LineID
LineItems
PK,FK1,FK4 LineID
FK3
FK2,I1
Describe
BaseID
SubUnit
CoopID
IIAACode
CDPLine
CountMe
TypeDuration
ProdMult
DefaultFIL
UMCFixed
UMCPerMile
UMCReload
UMCPeopleFact
BasePayinUMC
DrawPers
Speed
GetAway
AddWalkIn
LineCutters
LeaveBehind
UseAutoMRT
Comment
ImageFile
NeedsWater
StartDate
EndDate
DaysEff
PersCount
ParentID
ChildID
SquadCount
SquadDelay
SquadReturn
ChildDelay
SubUnits
PK Code
Describe
OST
PK,FK1,FK2 LineID
PK,FK1,I1 OptID
PK,FK2,I1 FMZCode
PK,FK2,I1 RL
I1 FIL
PRates
PK LineIDPart
PK NFDRSFM
PRate
MRT
PK LineID
PK,I1 FMZCode
PK,I1 RL
UMC
ArvTm
PRate
TypeDuration
ReloadDelay
ReloadPRate
UMCReload
IIAACodes
PK CodeSet
PK IIAACode
Describe
LineIDPart
PK,FK1 LineIDPart
PK,FK1 NFDRSFM
Describe
CDPLine
TypeDuration
UMCFixed
UMCPerMile
UMCReload
Speed
GetAway
AddWalkIn
LineCutters
LeaveBehind
DefaultFIL
NOTE: LineIDPart supplies default values for resource groups during the creation or edit of Line Items or the MRT
Coop
PK CoopID
Describe
CoopATGetAway
CoopXGetAway
PRateFactor

Miscellaneous Tables

Param	
PK	**ParamCode**
	ParamVal ParamText

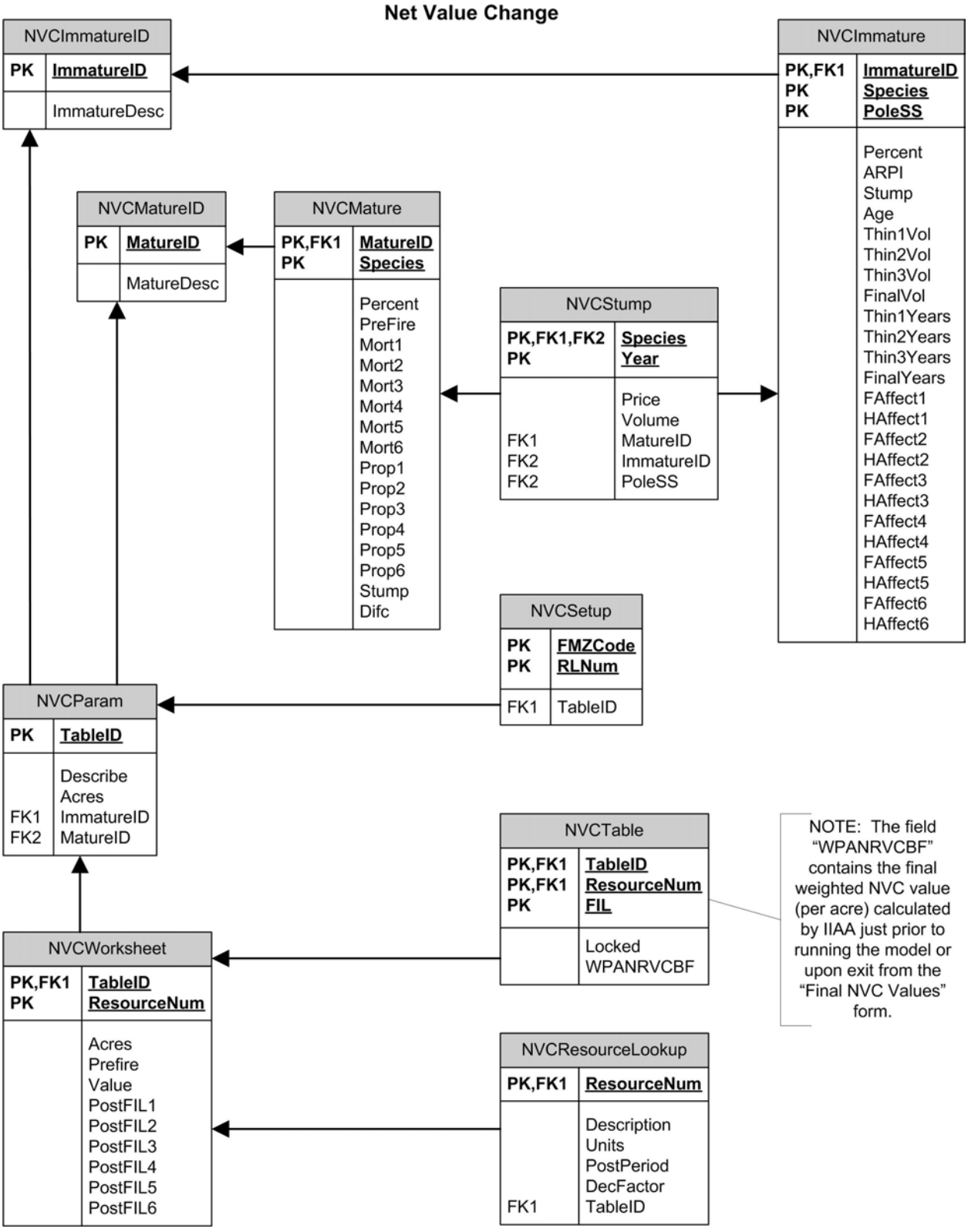
ListReport	
PK	**RepName**
	Local SQLText

Model Output Tables

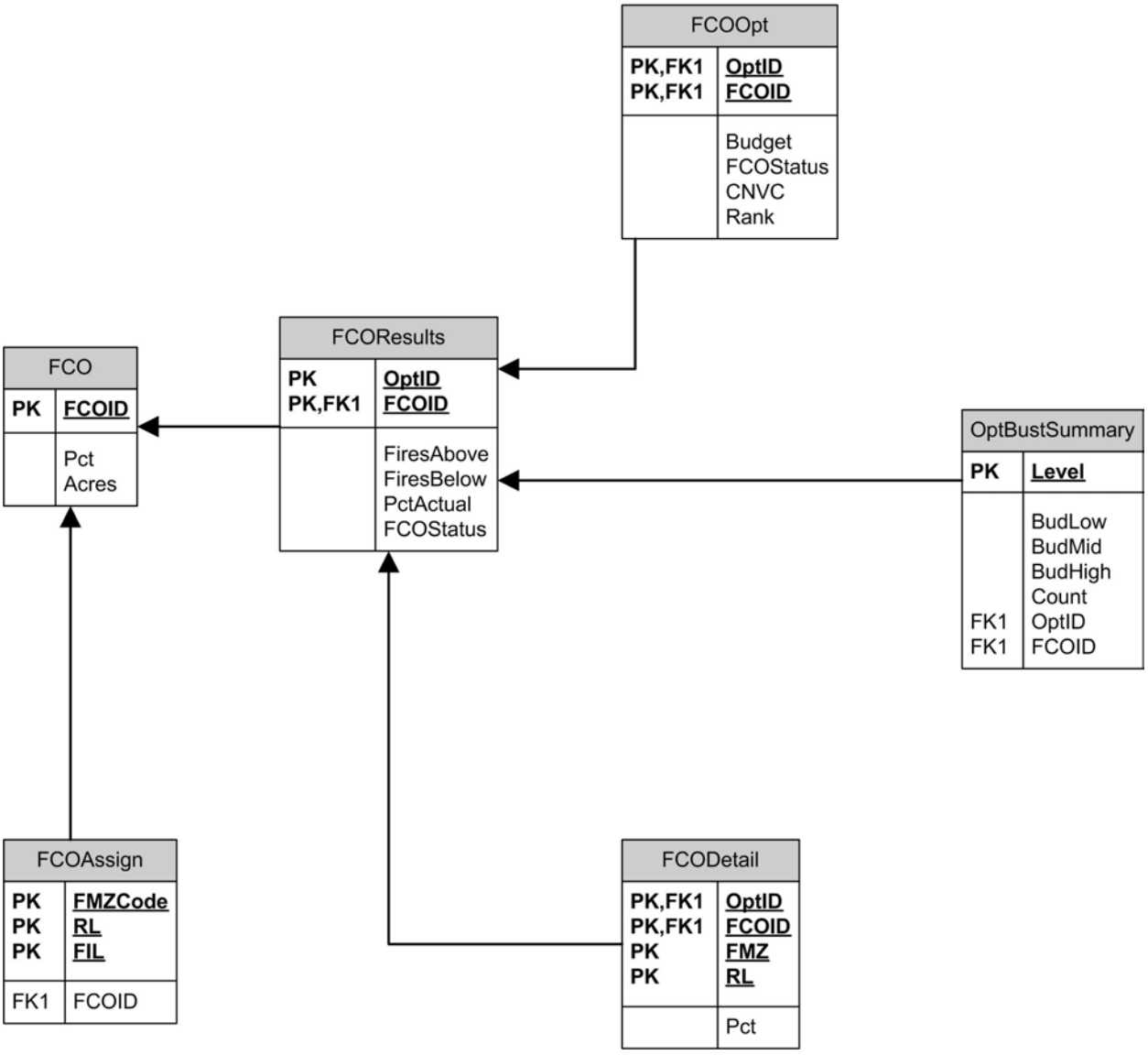
Runs	
PK **PK,I1** **PK,I1**	**FMZCode** **OptID** **FBDCode**
I1	RunNum Date

RunTable14	
PK,FK1,I1 **PK** **PK** **PK**	**RunNum** **RL** **FIL** **ROS5090**
	Freq Status Acres Chains LineBuilt UMC AAC NVC Minutes UMCAgency

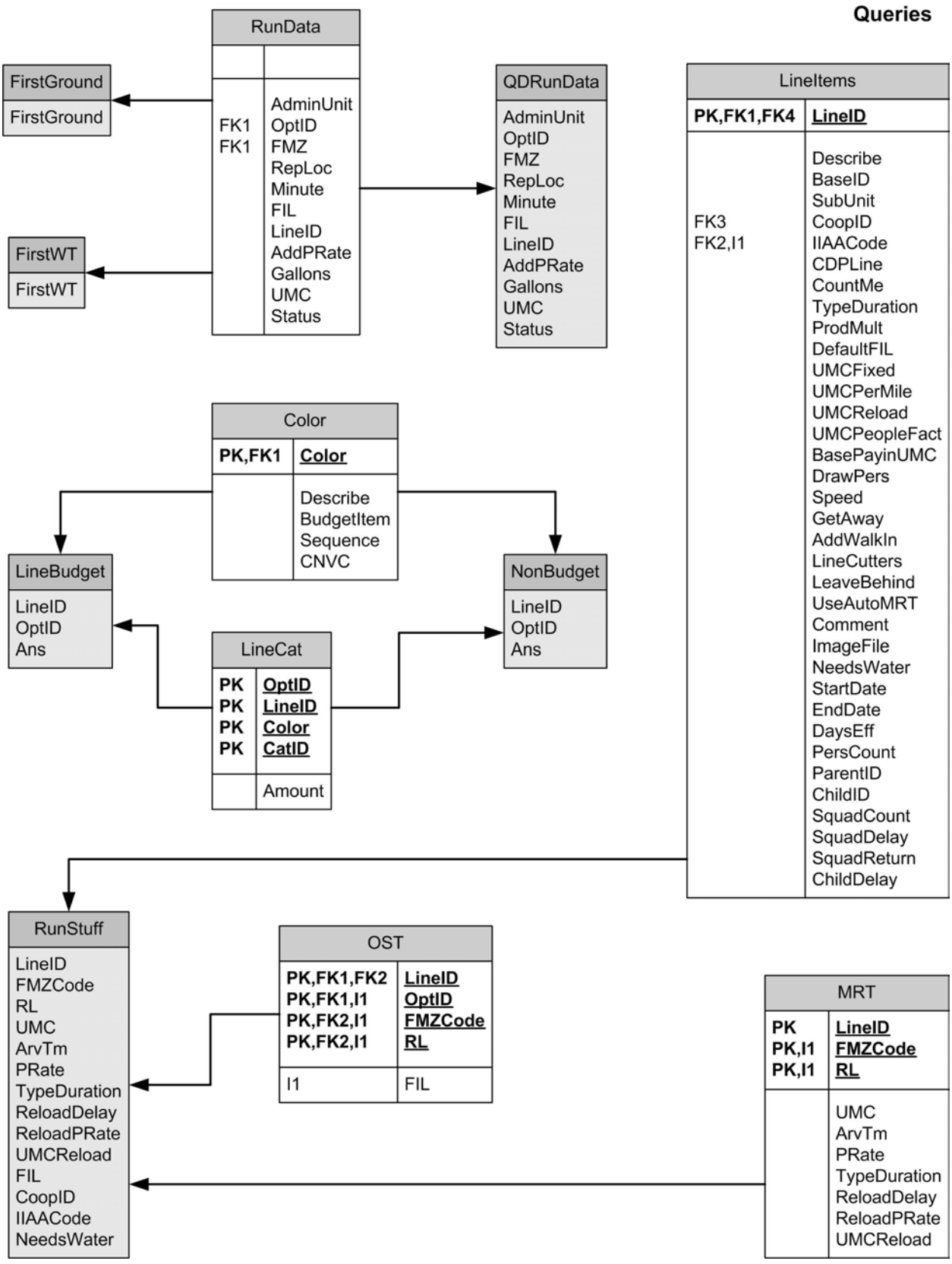
Net Value Change

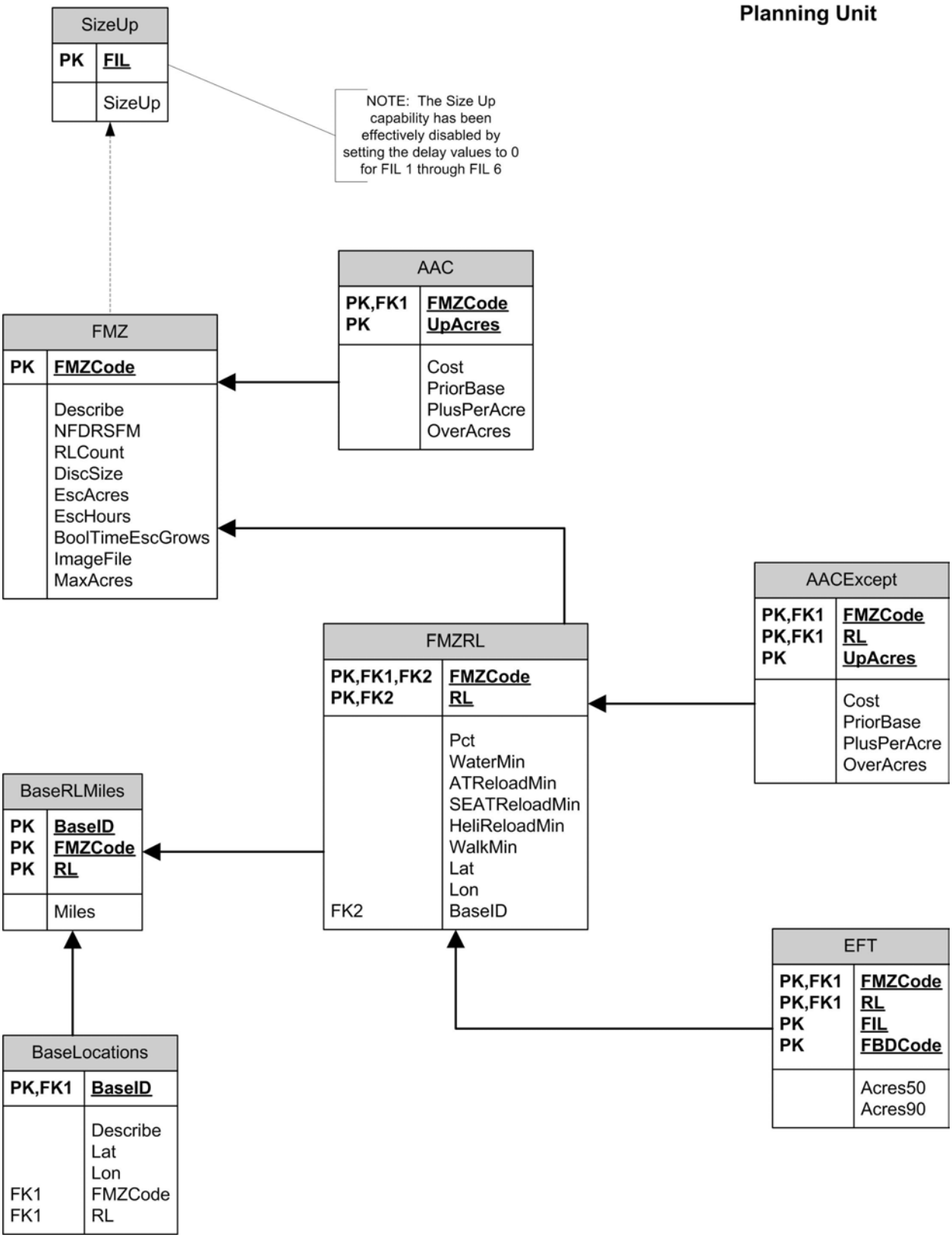


Option Buster



Queries


RunData
AdminUnit
FK1 OptID
FK1 FMZ
RepLoc
Minute
FIL
LineID
AddPRate
Gallons
UMC
Status
FirstGround
FirstGround
FirstWT
FirstWT
QDRunData
AdminUnit
OptID
FMZ
RepLoc
Minute
FIL
LineID
AddPRate
Gallons
UMC
Status
LineItems
PK,FK1,FK4 LineID
Describe
BaseID
SubUnit
FK3 CoopID
FK2,I1 IIAACode
CDPLine
CountMe
TypeDuration
ProdMult
DefaultFIL
UMCFixed
UMCPerMile
UMCReload
UMCPeopleFact
BasePayinUMC
DrawPers
Speed
GetAway
AddWalkIn
LineCutters
LeaveBehind
UseAutoMRT
Comment
ImageFile
NeedsWater
StartDate
EndDate
DaysEff
PersCount
ParentID
ChildID
SquadCount
SquadDelay
SquadReturn
ChildDelay
Color
PK,FK1 Color
Describe
BudgetItem
Sequence
CNVC
LineBudget
LineID
OptID
Ans
NonBudget
LineID
OptID
Ans
LineCat
PK OptID
PK LineID
PK Color
PK CatID
Amount
RunStuff
LineID
FMZCode
RL
UMC
ArvTm
PRate
TypeDuration
ReloadDelay
ReloadPRate
UMCReload
FIL
CoopID
IIAACode
NeedsWater
OST
PK,FK1,FK2 LineID
PK,FK1,I1 OptID
PK,FK2,I1 FMZCode
PK,FK2,I1 RL
I1 FIL
MRT
PK LineID
PK,I1 FMZCode
PK,I1 RL
UMC
ArvTm
PRate
TypeDuration
ReloadDelay
ReloadPRate
UMCReload


Planning Unit
SizeUp
PK FIL
SizeUp
NOTE: The Size Up capability has been effectively disabled by setting the delay values to 0 for FIL 1 through FIL 6
AAC
PK,FK1 FMZCode
PK UpAcres
Cost
PriorBase
PlusPerAcre
OverAcres
FMZ
PK FMZCode
Describe
NFDRSFM
RLCount
DiscSize
EscAcres
EscHours
BoolTimeEscGrows
ImageFile
MaxAcres
AACExcept
PK,FK1 FMZCode
PK,FK1 RL
PK UpAcres
Cost
PriorBase
PlusPerAcre
OverAcres
FMZRL
PK,FK1,FK2 FMZCode
PK,FK2 RL
Pct
WaterMin
ATReloadMin
SEATReloadMin
HeliReloadMin
WalkMin
Lat
Lon
FK2 BaseID
BaseRLMiles
PK BaseID
PK FMZCode
PK RL
Miles
EFT
PK,FK1 FMZCode
PK,FK1 RL
PK FIL
PK FBDCode
Acres50
Acres90
BaseLocations
PK,FK1 BaseID
Describe
Lat
Lon
FK1 FMZCode
FK1 RL